

Introduction

Fieldbus is not a replacement for 4-20 mA or Intelligent/Smart Transmitter Protocols, it provides much more. Fieldbus is a complete Control System Architecture enabling distribution of the control function to equipment in the field, therefore it is a replacement for the DCS Architecture of the 1970s.

To achieve the desired control the devices must be configured. That includes calibration but also building of a control strategy. The latter is covered in this manual.

One of the major advantages of Fieldbus is interoperability. Some blocks described in this manual are used not only by Smar devices , but other Foundation Fieldbus devices too. No particular configuration tool is addressed in this manual, because the devices are independent of configuration tool due to the DD technology.

Get the best results of the Fieldbus System by carefully reading these instructions.

Table of Contents

INTRODUCTION.....	I
TABLE OF CONTENTS	1
1 – INTRODUCTION TO FUNCTION BLOCK APPLICATION.....	5
OVERVIEW.....	5
FUNCTION BLOCK	5
TRANSDUCER BLOCK.....	5
RESOURCE BLOCK	5
FUNCTION BLOCK DEFINITIONS.....	5
FUNCTION BLOCK LINKAGES	5
INFORMATION ACCESS.....	6
FUNCTION BLOCK APPLICATION STRUCTURE	7
BLOCK OBJECT	7
BLOCK PARAMETERS.....	7
PARAMETER IDENTIFIERS.....	8
PARAMETER STORAGE.....	8
PARAMETER USAGE	8
Contained.....	8
Output.....	8
Input.....	9
PARAMETER RELATIONSHIPS.....	9
PARAMETER STATUS.....	9
COMPOSITION OF STATUS	10
PROCESS VARIABLE CALCULATION	13
SETPOINT CALCULATION	13
OUTPUT CALCULATION	14
CASCADE CONTROL	14
MODE PARAMETER.....	16
SCALING PARAMETERS.....	24
FAULT STATE HANDLING.....	25
FAULT STATE ACTIVE	26
ALARMS AND EVENTS – ALERT PROCESSING	29
SIMULATION.....	35
CHANNEL CONFIGURATION.....	36
BLOCK INSTANTIATION	37
ORDER OF PARAMETERS DURING DOWNLOAD	37
DATA TYPE AND DATA STRUCTURE DEFINITION.....	38
Block Structure – DS-64.....	39
Value & Status - Floating Point Structure – DS-65	39
Value & Status - Discrete Structure – DS-66	40
Scaling Structure – DS-68.....	40
Mode Structure – DS-69.....	40
Access Permissions – DS-70	41
Alarm Float Structure – DS-71	41
Alarm Discrete Structure – DS-72.....	41
Event Update Structure – DS-73.....	41
Alarm Summary Structure – DS-74	42

Simulate - Floating Point Structure – DS-82.....	42
Simulate - Discrete Structure – DS-83.....	42
Test Structure – DS-85.....	43
MANUFACTURER SPECIFIC DATA STRUCTURE	44
Scaling Conversion Structure - DS-256.....	44
Scaling Conversion Structure with Status - DS-257	44
Scaling Locator Structure - DS-258	44
Scaling Locator Structure with Status- DS-259	44
Modbus Variable Locator Structure - DS-260.....	45
Modbus Variable Locator Structure with Status- DS-261.....	45
FF Parameter ID Structure - DS-262	45
Slave Address Structure - DS-263.....	45
2 – BLOCK LIBRARY.....	47
2.1 - DESCRIPTION OF BLOCK TYPES	47
2.2 - BLOCK TYPE AVAILABILITY AND INITIAL BLOCK SET	51
2.3 - RESOURCE	53
RS – Resource Block	53
2.4 - TRANSDUCER BLOCKS	59
DIAG – Diagnostics Transducer Block	59
Display Transducer.....	62
HC – Hardware Configuration Transducer.....	64
MBCF – ModBus Configuration.....	68
2.5 - INPUT TRANSDUCER BLOCKS	69
LD292 / LD302 - Pressure Transducer	69
TT302 - Temperature Transducer.....	70
IF302 - Current Fieldbus Transducer	72
TP302 – Position Fieldbus Transducer	73
TEMP – DF-45 Temperature Transducer	74
2.6 - OUTPUT TRANSDUCER BLOCKS.....	80
FY302 – Fieldbus Positioner Transducer	80
FP302 - Fieldbus Pressure Transducer.....	82
FI302 – Fieldbus Current Transducer.....	84
2.7 - INPUT FUNCTION BLOCKS	85
AI - Analog Input.....	85
DI - Discrete Input.....	91
MAI - Multiple Analog Input	95
MDI - Multiple Discrete Input.....	98
PUL – Pulse Input	101
2.8 - CONTROL AND CALCULATION FUNCTION BLOCKS	107
PID - Pid Control	107
APID – Advanced Pid	119
ARTH - Arithmetic.....	125
SPLT-Splitter	134
CHAR - Signal Characterizer.....	139
INTG - Integrator	142
AALM - Analog Alarm	150
ISEL - Input Selector.....	158
SPG - Sepoint Ramp Generator.....	162
TIME – Timer and Logic.....	169
LLAG - Lead Lag	178
OSDL - Output Signal Selector and Dynamic Limiter	181
DENS - Density	186
CT – Constant	191
FFET - Flip-Flop and Edge Trigger	194
MBCS – ModBus Control Slave.....	197

MBSS – ModBus Supervision Slave	201
MBCM – ModBus Control Master	203
MBSM – ModBus Supervision Master	207
2.9 - OUTPUT FUNCTION BLOCKS	209
AO - Analog Output	209
DO - Discrete Output	214
MAO - Multiple Analog Output	218
MDO - Multiple Discrete Output	222
STEP – Step Output Pid	225
BLOCK OPTIONS	234
Resource Block Bit Strings	234
FUNCTION BLOCK OPTIONS	237
IO_OPTS	237
CONTROL_OPTS	239
STATUS_OPTS	241
ALARM_SUM and ACK_OPTION	242
PID_OPTS	242
INTEG_OPTS	244
INVERT_OPTS	244
INPUT_OPTS	245
OSDL_OPTS	246
MO_STATUS_OPTS	246
MO_OPTS (Profile Rev. 0 – FB700)	247
MO_OPTS (Profile Rev. 1 – DFI302)	247
MODULE_STATUS_RO_3	248
MODULE_STATUS_R4_7	248
MODULE_STATUS_R8_11	250
MODULE_STATUS_R12_14	250
3 - EXAMPLES	252
SIMPLE CONTROL APPLICATION	252
CASCADE CONTROL	253
Corresponding Configuration	254
Parameterization	255
RATIO CONTROL	256
Corresponding Configuration	257
Parameterization	257
FEEDFORWARD CONTROL	259
Corresponding Configuration	259
Parameterization	260
SPLIT RANGE CONTROL	261
Corresponding Configuration	262
Parameterization	263
LEVEL CONTROL	264
Corresponding Configuration	265
Parameterization	265
RATE CONTROL LOOP WITH LEAD-LAG	267
Corresponding Configuration	268
Parameterization	268
FLOW COMPENSATION CONFIGURATION, WITH TOTALIZATION	270
Corresponding Configuration	271
Parameterization	271
HYDROSTATIC TANK GAUGING	273
Corresponding Configuration	274
Parameterization	274
COMBUSTION CONTROL WITH DOUBLE CROSS LIMITS	276

<i>Corresponding Configuration</i>	277
<i>Parameterization</i>	277
3 ELEMENT BOILER LEVEL / FEEDWATER CONTROL.....	279
<i>Corresponding Configuration</i>	280
<i>Parameterization</i>	280

1 – Introduction to Function Block Application

Overview

Function block applications are defined as plant or factory applications that perform one or more automatic monitoring and control functions.

Function Block

Function blocks represent the basic automation functions performed by the function block application. Each function block processes input parameters according to a specified algorithm and an internal set of control parameters. They produce output parameters that are available for use within the same function block application or by other function block applications.

Transducer Block

Transducer blocks insulate function blocks from the specifics of I/O devices, such as sensors, actuators, and switches. Transducer blocks control access to I/O devices through a device independent interface defined for use by function blocks. Transducer blocks also perform functions, such as calibration and linearization, on I/O data to convert it to a device independent representation. Their interface to function blocks is defined as one or more implementation independent I/O channels.

Resource Block

Resource blocks are used to define hardware specific characteristics of function block applications. Similar to transducer blocks, they insulate function blocks from the physical hardware by containing a set of implementation independent hardware parameters.

Function Block Definitions

Function blocks are defined by their inputs, outputs, control parameters, and by the algorithm that operates on these parameters. Function blocks are identified using a name (Tag) and a numeric index.

Tags provide a symbolic reference to function blocks. They are unambiguous within the scope of a fieldbus system. Numeric indices are

numbers assigned to optimize access to function blocks. As opposed to function block tags, which are global, numeric indices have meaning only within the application that contains the function block.

Function block parameters define the inputs, outputs, and the data used to control function block operation. They are visible and accessible over the network. Additional parameters, called “contained within” parameters are used to define the private data of a function block. Although visible over the network, they may not participate in function block linkages.

Function Block Linkages

Function block outputs may be linked to inputs of other function blocks. Each linkage indicates that an input parameter of one function block obtains its value from a specific output parameters of another function block. While function block “pull” their values from upstream blocks, which block controls the “pulling” depends on the characteristics of the underlying communications.

Two function blocks linked together may exist in the same function block application, or in separate applications, either in the same device or in different devices.

To transfer the data for a function block link, the communication channel must be known, that provide the transfer of parameter data (and other types of data) between applications.

Information Access

Function block information may be grouped for access depending on how it is to be used. The following four groups are defined for access purposes:

- 1) dynamic operation data,
- 2) static operation data,
- 3) all dynamic data, and
- 4) other static data.

To support access of operator interface information during function block execution, two levels of network access are defined, one for operational traffic and one for background traffic. Operator interface traffic is transferred as background traffic to prevent it from interfering with the operation of time-critical function blocks.

Function Block Application Structure

Function block applications are modeled as a set of function blocks coordinated to execute a related set of operations. This set of operations collectively provides a single, higher level control function.

Function block model is real-time algorithm that transforms input parameters into output parameters. Their operation is controlled through the setting of control parameters.

Interoperation between function blocks is modeled by linking an input parameter of one function block to an output parameter of another. Function blocks can be bound together within and across devices. Interfaces between function blocks located in the same function block application are locally defined. Those interfaces between function blocks in different devices use the communication services.

To support function block operation, the function block architecture also provides transducer and resource blocks, and display objects.

Function Block Application Process represents the function block application as an integrated set of these components accessed to its network interface.

Block Object

A block object represents a logical processing unit composed of a set of input, processing, and control parameters and an associated algorithm.

Each block is identified by its Tag which is defined to be unique throughout the control system at one plant site. Block tags are defined as strings with a maximum length of 32 characters.

During system operation, a short hand reference, known as a numeric index is used for block access purposes. A block's numeric index is unique only within the function block application where it exists.

The algorithm of a block is identified by its type and the revision level of its type. This information indicates how the execution of the algorithm is affected by control parameters.

Block Parameters

Parameters define the inputs, outputs, and control data for a block. Their relationship to each other and to the block algorithm is shown below.

Parameter Identifiers

Each parameter has a name composed by 4 bytes unsigned integer machine. Parameter names are unique within a block. Within a system, a parameter can be unambiguously identified by qualifying its name with the tag of its block. This construction is referred to as "Tag.Parameter".

The Tag.Parameter construct is used to obtain the index of a parameter. This is the second way of identifying a parameter.

Parameter Storage

Parameter attributes may be classified as dynamic, static, or non-volatile. The value of parameter attributes may need to be restored after a power failure based on its classification:

Dynamic - a parameter attribute whose value is calculated by the block algorithm and therefore does not need to be restored after a power failure.

Static - a parameter attribute which has a specific configured value that must be restored by a device after power failure. An Interface or temporary devices may write to static parameter attributes on an infrequent basis. Static parameter attribute values are normally tracked by a configuration device. To support tracking changes in static parameter attributes, the associated block's static revision parameter will be incremented and an update event will be generated each time a static parameter attribute value is modified.

Non-volatile - a parameter attribute whose value is written on a frequent basis and the last saved value must be restored by the device after a power failure. Since the values of these parameter attributes are constantly changing, they are not normally tracked by a configuration device.

The classification of a parameter attribute will determine the manner in which the attribute value is stored within a device.

Parameter Usage

Parameters are defined for a block for a specific purpose. Each is defined for use as an input, an output, or a control parameter. Control parameters are also referred to as "contained" parameters because they may not be linked with parameters in other blocks. Each type of usage is defined as follows:

Contained

A contained parameter is a parameter whose value is configured, set by an operator, higher level device, or calculated. It may not be linked to another function block input or output. The mode parameter is an example of a contained parameter common to all blocks.

Output

An output parameter is a parameter that may be linked to an input parameter of another function block. Output parameters contain status. The output status indicates the quality of the parameter value and the mode of the block when it was generated.

The value of an output parameter may not be obtained from a source external to the block. It may be generated by the block algorithm, but does not have to be.

The values of certain output parameters are dependent on the value of the mode parameter of the block. These output parameters may be referred to as mode-controlled output parameters.

Blocks whose purpose is to generate a single output contain one parameter designed as the primary output parameter. Primary outputs are used by other blocks for control or calculation purposes. These blocks also contain secondary output parameters such as alarm and event parameters that play a supporting role to the primary output parameter.

Input

An input parameter obtains its value from a source external to the block. An input parameter may be linked to an output parameter of another function block. Its value may be used by the algorithm of the block.

Input parameter values are accompanied by status. When an input parameter is linked to an output parameter, the status will be provided as the status of the output parameter. When it is not linked to an output parameter, the status will indicate that the value was not provided by an output parameter. When an expected input parameter value is not received, the function block supported services responsible for delivering the data will set the status of the input parameter to indicate the failure.

If an input parameter is not linked to an output parameter, then it will be treated as a constant value by the function block application. The difference between unlinked input parameters and contained parameters is that input parameters have the capability to support a linkage and contained parameters do not.

Blocks whose purpose is to transform or operate on a single input will contain one parameter designed as the primary input parameter. One input parameter of some types of blocks is designated as the primary input parameter. Primary inputs are used for control or calculation purposes. These blocks may also contain secondary input parameters that support processing done on the primary input parameter.

Parameter Relationships

The execution of a block involves the inputs, outputs, contained parameters, and the algorithm of the block. The execution time for a block's algorithm is defined as a parameter of the block. Its value is dependent on how the block was implemented.

The input parameters are used by the algorithm in conjunction with the state of the function block application containing the block to determine if the algorithm can achieve the target mode established for it. The target mode is the part of the mode parameter that indicates what mode of operation is desired for the block. It is normally set by a control device or the operator.

Under certain operating condition a block may not be able to function in the requested mode. In such cases, the actual mode reflects the mode it is able to achieve. Comparison of the actual against the target indicates whether the target was achieved.

The values for the mode parameter for a block are defined by the Permitted Mode parameter. Thus, the modes available for controlling a block may vary with each block.

The values assigned to the Permitted Mode are selected from those defined by the block designer. They are assigned during block configuration for the specific use of the function block application.

Once the actual mode is determined, the block execution progresses and the outputs are generated.

Parameter Status

All input and output parameters are structures composed of status and value, but some contained parameter (internal parameter, not accessible by other blocks) have also that data type, for example, RCAS_IN, ROUT_IN, SP and PV.

The Status field is composed of three parts: Quality, Sub-Status and Limits.

Quality – It indicates the quality of the parameter value.

Good Cascade – The quality of the value is good, and it may be part of a cascade structure.

Good Non-Cascade – The quality of the value is good, and the block doesn't support a cascade path.

Uncertain – The quality of the value is less than normal, but the value may still be useful.

Bad – The value is not useful.

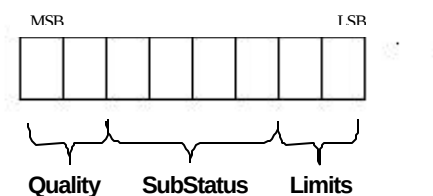
Sub-Status – The sub-status is a complement of the quality status and takes information to initialize or break a cascade control, alarms and others. There are different sets of sub-status for each quality.

Limits – It provides information whether the associated value is limited or not, as well the direction. The limits are classified as : Not Limited, High Limited, Low Limited, Constant.

When an input parameter is linked to an output parameter through the link object, the whole structure (status and value) is copied (local link) or received from the bus (external link). If the input is not linked, then the status may be set manually by the user, as well the value.

Composition of Status

The Status has the following composition:



The quality, sub-status, and limit components of status are defined as follows:

Quality - The quality used will be determined by the highest priority condition:

- 0 = Bad
- 1 = Uncertain
- 2 = Good (Non-cascade)
- 3 = Good (Cascade)

Substatus - Sub-status values in the status attribute are defined as shown in the following table.

Limit - The following limit conditions will be always available in the status attribute.

- 0 = Not limited
- 1 = Low limited
- 2 = High limited
- 3 = Constant

Examples:

0xC1 (in hexadecimal) is "Good-Cascade Non Specific and Low Limited " status

0xCF(in hexadecimal) is "Good-Cascade Not invited and Constant " status

0x4E(in hexadecimal) is "Uncertain Initial Value and High Limited" status

Quality	Sub-status	Hex value	Not in cascade	Forward path of cascade	Backward path of cascade
Bad	0 = Non-specific	0x00	X	X	X
Bad	1 = Configuration Error	0x04	X	X	X
Bad	2 = Not Connected	0x08			
Bad	3 = Device Failure	0x0c	X	X	X
Bad	4 = Sensor Failure	0x10	X	X	X
Bad	5 = No Communication, with last usable value	0x14			
Bad	6 = No Communication, with no usable value	0x18			
Bad	7 = Out of Service (highest priority)	0x1c			

Quality	Sub-status	Hex value	Not in cascade	Forward path of cascade	Backward path of cascade
Uncertain	0 = Non-specific	0x40	X		
Uncertain	1 = Last Usable Value	0x44	X		
Uncertain	2 = Substitute	0x48	X		
Uncertain	3 = Initial Value	0x4c	X		
Uncertain	4 = Sensor Conversion not Accurate	0x50	X		
Uncertain	5 = Engineering Unit Range Violation	0x54	X		
Uncertain	6 = Sub-normal	0x58	X		

Quality	Sub-status	Hex value	Not in cascade	Forward path of cascade	Backward path of cascade
GoodNC	0 = Non-specific (lowest priority)	0x80	X		
GoodNC	1 = Active Block Alarm	0x84	X		
GoodNC	2 = Active Advisory Alarm	0x88	X		
GoodNC	3 = Active Critical Alarm	0x8c	X		
GoodNC	4 = Unacknowledged Block Alarm	0x90	X		
GoodNC	5 = Unacknowledged Advisory Alarm	0x94	X		
GoodNC	6 = Unacknowledged Critical Alarm	0x98	X		

Quality	Sub-status	Hex value	Not in cascade	Forward path of cascade	Backward path of cascade
GoodC	0 = Non-specific	0xc0		X	X
GoodC	1 = Initialization Acknowledge(IA)	0xc4		X	
GoodC	2 = Initialization Request(IR)	0xc8			X
GoodC	3 = Not Invited (NI)	0xcc			X
GoodC	4 = Not Selected(NS)	0xd0			X
GoodC	6 = Local Override(LO)	0xd8			X
GoodC	7 = Fault State Active(FSA)	0xdc			X
GoodC	8 = Initiate Fault State (IFS)	0xe0		X	

Example : conversion from the enumerations to number

The following formula is used to obtain the enumeration number of a determinate status attribute:

$$\text{Decimal Value Status} = 64 * \text{Quality} + 4 * \text{Sub-Status} + \text{Limit}$$

For example, considering the following status:

“Uncertain - Initial Value - High Limited”

where:

Quality = “uncertain” = 1

Sub-Status = “Initial Value” = 3

Limit = “High Limited” = 2

Applying the formula:

Decimal Value Status = $64 * 1 + 4 * 3 + 2 = 78$ (in decimal) or 0x4E (in Hexadecimal)

Example: conversion from number to enumerations

There are many forms to convert the enumerate number to the status string. Below is shown two forms to do this.

Expressing the number in binary.

Hex Value Status = 78 = 0x4E = 01001110 (in binary)

Dividing this binary number in quality, sub-status and limit fields:

Quality = 01 = 1 = “Uncertain”

Sub-Status = 0011 = 3 = “Initial Value”

Limit = 10 = 2 = “High Limited”

The corresponding status is “Uncertain - Initial Value - High Limited”.

Using the value of status in decimal format.

Decimal Value Status = 78

Divided the number by 64. The quotient will be the Quality and save the remainder:

Quality = $78 / 64 = 1$

Remainder = 14

Divided the remainder by 4. The quotient will be the Sub-Status and the remainder will be the limit:

SubStatus = $14 / 4 = 3$

Limit = 2

Process Variable Calculation

The process variable (PV) parameter reflects the value and status of the primary input value or calculated value based on multiple input. The PV parameter is the IN parameter after filtering (PID and AALM), or it reflects the value from the transducer after filtering (AI and AO-readback), or the combination of two input parameters for range extension (ARTH).

The PV parameter has a status, although it is a contained parameter. This status is a copy of the primary input status or the worst status when the PV is based on multiple inputs. The PV value reflects the value of the calculated input regardless the mode of the block, unless this input is not usable, then the PV holds the last usable value.

Optionally, a filter may be applied in the process value signal, whose time constant is PV_FTIME. Considering a step change to the input, this is the time in seconds to the PV reaches 63,2% of the final value. If the PV_FTIME value is zero, the filter is disabled.

Setpoint Calculation

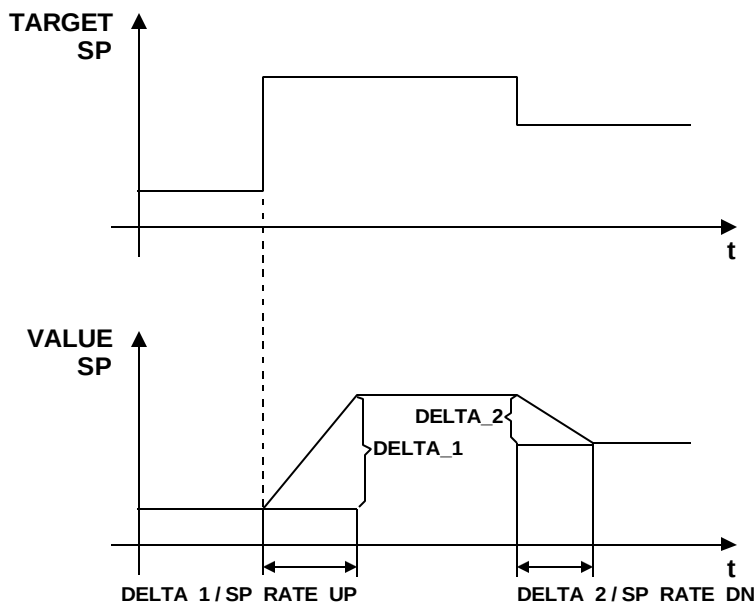
SP limits : SP_HI_LIM and SP_LO_LIM

At first, the SP will be limited to a range specified by the SP_HI_LIM and SP_LO_LIM parameters **only in Auto mode**. However in the PID block, if the bit "Obey SP limits if Cas or Rcas" in CONTROL_OPTS parameter is true, then the setpoint value will be also restricted to setpoint limits in Cas and RCas mode.

SP rate limits: SP_RATE_UP and SP_RATE_DN

These parameters avoid bump in SP change, and they depend on the block type as well the mode to be effective. The SP rate limits are applied for the PID block in Auto mode, and AO block in Auto, Cas or RCas modes.

When the block is in Auto mode and the user change the SP to a value greater than the current value, then the SP value ramps upward based on the SP_RATE_UP parameter. If the new setpoint is less than the current value, the SP value ramps downward based on the SP_RATE_DN. When the SP_RATE_DN and/or SP_RATE_UP is zero the rate limiting is disabled.



The below table summarizes the conditions for SP limits and SP rate limits.

Block type	Mode	Required configuration for SP limits (SP_HI_LIM/SP_LO_LIM)	Required configuration for SP rate limits (SP_RATE_UP/SP_RATE_DN)
PID	Auto	None	SP_RATE_UP / SP_RATE_DN different of zero
	Cas/Rcas	CONTROL_OPTS.“Obey SP limits if Cas or Rcas” is true	Not apply
AO	Auto	None	SP_RATE_UP / SP_RATE_DN different of zero
	Cas/Rcas	Not apply	SP_RATE_UP / SP_RATE_DN different of zero.

c) SP tracking PV

Some control strategies require that the transition from a “manual” mode (Rout, Man, LO and Iman) to an “auto” mode (Auto, Cas, Rcas) must be done with error equals to zero, therefore SP must be equal to PV.

The CONTROL_OPTS of the PID block and the IO_OPTS of the AO block may be configured for SP tracking PV when the block is in a “manual” mode.

This option is summarized in the following table:

BitString	CONTROL_OPTS (PID)	IO_OPTS (AO)	Meaning
SP-PV Track in Man	X	X	The SP tracks the PV when the target mode is Man.
SP-PV Track in Rout	X		The SP tracks the PV when the actual mode is Rout.
SP-PV Track in LO or Iman	X	X	The SP tracks the PV when the actual mode is LO or Iman.

Output Calculation

When the actual mode is AUTO, CAS or RCAS, the normal algorithm is executed. This calculation is specific for each function block type. If the mode is a “manual” mode, the output is just following a value provided by another block (LO,Iman), the user (Man) or a Control Application running on an interface device (Rout).

The output value is limited high and low by the OUT_HI_LIM and OUT_LO_LIM parameters in PID and ARTH blocks for all modes.

It is possible to disable the output limits in Manual mode in the PID block by setting “No OUT limits in Manual” bit in CONTROL_OPTS.

Cascade control

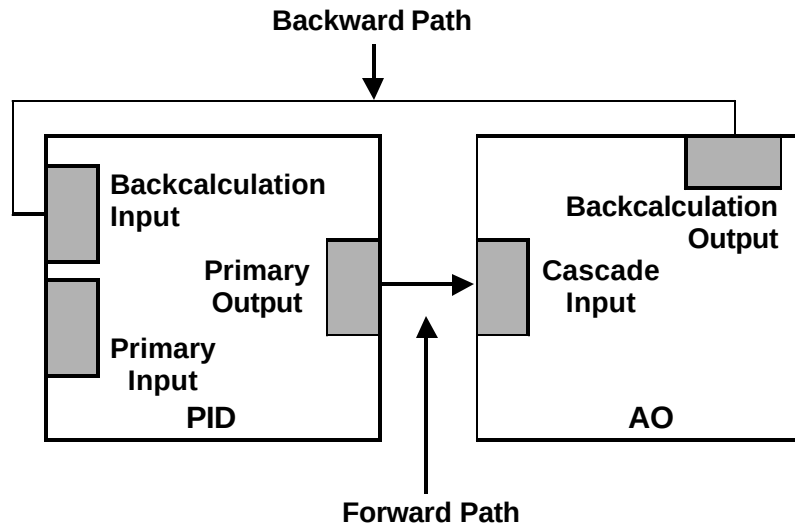
There is a linked output and input pair involved in each of the different forms of cascade, as shown in the following table.

Mode	Forward	Backward
Cas	CAS_IN	BKCAL_OUT
RCas	RCAS_IN	RCAS_OUT
ROut	ROUT_IN	ROUT_OUT

Table 1 – Parameter Pairs

In a cascade, the upper control block provides an output value and status, which becomes the cascade input to the lower block.
The lower block in the cascade provides an output value, which is communicated to the upper block as back-calculation input.

Based on the following example, which is the most common form of cascade, it will be shown the process of cascade initialization.



There are four steps to complete a cascade initialization:

Not cascade mode – As the AO block is in Auto mode, the PID block is not calculating the output (OUT), it is just following the backward value (AO.BKCAL_OUT -> PID.BKCAL_IN).

PID
MODE_BLK.Target = Auto
MODE_BLK.Actual = IMan
OUT.Status = GoodC-Non-specific

AO
MODE_BLK.Target = Auto
MODE_BLK.Actual = Auto
BKCAL_OUT.Status = GoodC-Not Invited

Initialize – The user changes the target mode of AO block to Cas, then the AO block sets GoodC-IR in BKCAL_OUT. The value of BKCAL_OUT is the initial value for the PID starts to calculate. The AO block waits for the PID to set GoodC-IA in OUT, which is linked to AO.CAS_IN (PID.OUT -> AO.CAS_IN).

PID
MODE_BLK.Target = Auto
MODE_BLK.Actual = IMan
OUT.Status = GoodC-Non-specific

AO
MODE_BLK.Target = Cas
MODE_BLK.Actual = Auto
BKCAL_OUT.Status = GoodC-Initialization Request (IR)

Initialization complete – The AO block goes to Cas, because the PID block sent GoodC-IA.

PID
MODE_BLK.Target = Auto

MODE_BLK.Actual = IMan
OUT.Status = GoodC- Initialization Acknowledge (IA)

AO
MODE_BLK.Target = Cas
MODE_BLK.Actual = Cas
BKCAL_OUT.Status = GoodC- Non-specific

4. **Cascade complete** – The PID block changes the status of OUT from GoodC-IA to GoodC-NS.

PID
MODE_BLK.Target = Auto
MODE_BLK.Actual = Auto
OUT.Status = GoodC- Non-specific

AO
MODE_BLK.Target = Cas
MODE_BLK.Actual = Cas
BKCAL_OUT.Status = GoodC- Non-specific

Notes:

- The remote cascade modes (Rcas and Rout) have a similar mechanism to the process of cascade initialization.
- The function blocks prepared to work as the upper block in the cascade have the BKCAL_IN parameter, as PID, SPLT, SPG and OSDL
- The function blocks prepared to work as the lower block in the cascade have the BKCAL_OUT parameter, as PID, AO, SPLT and OSDL.
- The upper blocks will be in Iman mode when the lower block is not in cascade, whose main reasons are :
 - Link failure in backward path (lower.BKCAL_OUT -> upper.BKCAL_IN)
 - Lower block can not execute in Cas: the target mode of lower block is not Cas, or there is any condition forcing the lower block to a higher priority mode as fault state(AO in LO), link failure in the forward path (OUT -> CAS_IN).
 - Tracking (PID in LO), link failure in the primary input (PID in Man), and others. (See details in the section Mode Parameter).
 - Control Application running on as interface device works similarly an upper block in the remote cascade modes (Rcas and Rout).

Mode Parameter

a) Mode types

The operation of the block is summarized for each mode type as follows:

Out of Service (O/S):

The block is not being evaluated. The output is maintained at last value or, in the case of output class function blocks, the output may be maintained at an assigned Fault State value – last value or configured Fault State value. Setpoint is maintained at last value.

Initialization Manual (IMan):

This mode means that the downstream block is not in cascade (Cas mode), therefore the normal algorithm must not be executed and the block output will just follow an external tracking signal (BKCAL_IN) coming from the downstream block. This mode cannot be requested through the target mode.

Local Override (LO):

It applies to control block that supports a track input parameter, when the control block is LO, its output is tracking the TRK_VAL input parameter.

The LO mode also applies to output block when it is in fault state. This mode cannot be requested through the target mode.

Manual (Man):

The block output is not being calculated, although it may be limited. The operator may set directly the outputs of the block.

Automatic (Auto):

The normal algorithm calculates the block output. If the block has a setpoint, it will be used a local value that may be written by an operator through an interface device.

Cascade (Cas):

The setpoint comes from another block over a link (CAS_IN), therefore the operator cannot change the setpoint. The normal algorithm calculates the block output based on that setpoint. In order to achieve this mode, the algorithm uses the CAS_IN input and BKCAL_OUT output to establish the cascade with the upstream block in a bumpless way.

Remote Cascade (RCas):

The block setpoint is being set by a Control Application running on an interface device to the RCAS_IN parameter. The normal algorithm calculates the block output based on that setpoint, so the block running in Rcas works similarly a "lower block" in cascade. In order to achieve this mode, the block

algorithm uses the RCAS_IN and RCAS_OUT to **establish a relation like a cascade** with the interface device in a bumpless way. Therefore the Control Application works **similarly as an "upper block"**, but its algorithm is not synchronized to the schedule and neither does it use a link to transfer the setpoint to the block.

Remote Output (ROut):

The block output is being set by a Control Application running on an interface Device to the ROUT_IN parameter. In order to achieve this mode, the block algorithm uses the ROUT_IN and ROUT_OUT to **establish a relation like a cascade** with the interface device in a bumpless way. Therefore the Control Application works **similarly as an "upper block"**, but its algorithm is not synchronized to the schedule and neither does it use a link to transfer the output to the block. The block running in ROut works similarly a "lower block" in cascade.

Auto, Cas, and RCas are the "automatic" modes, that calculate the primary output using the normal algorithm. The "manual" modes are IMan, LO, Man, and ROut.

Mode type	Source of SP	Source of OUT
O/S	User	User
Iman	User	Other function block – following BKCAL_IN parameter
LO	PID / EPID / APID : User AO / DO : Fault state (last value or FSTATE_VAL)	PID / EPID / APID : Other function block – following TRK_VAL parameter AO / DO : Fault state (last value or FSTATE_VAL)
Man	User	User
Auto	User	Block algorithm
Cas	Other function block – following CAS_IN parameter	Block algorithm
Rcas	Control Application running on an interface device	Block algorithm
Rout	Block keeps last value	Control Application running on an interface device

b) Elements of MODE_BLK

The mode parameter (MODE_BLK) is defined in every function block. It is defined as having four elements:

1) Target - This is the mode requested by the operator. Only one mode from those allowed by the permitted mode parameter may be requested, that check will be done by the device.

2) Actual - This is the current mode of the block, which may differ from the target based on operating conditions and block configuration, as input parameter status and bypass configuration, for example. Its value is *always* calculated as part of block execution, therefore the user can not write in this attribute.

3) Permitted – It defines the modes that are allowed for an instance of the block. The permitted mode is configured based on the application

requirement. For example, if a PID block does not have link for CAS_IN, the Cas mode should not be permitted for that block. It is like a list of mode types selected from the supported modes.

4) Normal - This is the mode which the block should be set to during normal operating conditions. The normal attribute is used as a *reminder*. It does not affect the algorithm calculation.

Execution of a function or transducer block will be controlled through the mode parameter. The user sets the target mode, which indicates what mode of operation is desired for the block. Then, the algorithm evaluates if the block can be executed in the *requested mode* (target mode) *or the nearest higher priority mode possible*. The actual mode reflects the mode of block operation.

Other concepts of mode:

Retained target: When the target mode is O/S, MAN, RCAS or ROUT the target mode attribute may retain information about the previous target mode. This information may be used by the block in mode shedding and setpoint tracking. This feature is optional and the interface device is responsible to implement it.

Supported mode: Each *block type* has a set of mode types supported, it means that *the block definition specifies* in which modes the block may operate.

c) Priority of mode

The concept of priority is used when the block calculates the actual mode, and when determining if write access is allowed for a particular mode or other of higher priority.

Mode	Description	Priority
O/S	Out of Service	7 – highest
IMan	Initialization Manual	6
LO	Local Override	5
Man	Manual	4
Auto	Automatic	3
Cas	Cascade	2
Rcas	Remote Cascade	1
Rout	Remote Output	0 - lowest

Priority of the Mode

d) Mode shedding

Interface devices such as a host computer, distributed control system (DCS) controller, or programmable logic controller (PLC) may exist which do not support the function block application architecture but have proprietary control applications running on them.

Such applications may adjust the values of the block setpoint (RCas mode) and/or primary output (Rout mode) parameters in a function block. When doing so, they provide the value of each parameter along with its status.

If a new value is not received by the function block within a specified “update time” (SHED_RCAS and SHED_ROUT), or a bad status is received, then the function block mode will be changed to a non-remote mode of higher priority.

The SHED_OPT parameter configures the desired behavior when shedding from a remote mode (RCas and Rout), therefore **it does not include the Cas mode**. Also, it determines if the shed mode is maintained once the RCAS_IN or ROUT_IN parameter updating is recovered (no return – target mode receives the shed mode) or original mode will be restored when the cause of shedding has cleared (normal return – no change in target mode).

The shed option has the following enumeration:

- 0 = Undefined – Invalid
- 1 = Normal shed, normal return – Actual mode changes to the next lowest priority non-remote mode permitted but returns to the target remote mode when the remote computer completes the initialization handshake.
- 2 = Normal shed, no return – Target mode changes to the next lowest priority non-remote mode permitted. The target remote mode is lost, so there is no return to it.
- 3 = Shed to Auto, normal return.
- 4 = Shed to Auto, no return – Target mode changes to Auto on detection of a shed condition.
- 5 = Shed to Manual, normal return.
- 6 = Shed to Manual, no return – Target mode changes to Man on detection of a shed condition.
- 7 = Shed to Retained target, normal return.
- 8 = Shed to Retained target, no return (change target to retained target).

e) Mode calculation

The actual mode will be calculated based on the following:

Each mode type has some conditions that force the actual mode to be of higher priority than the target mode.

Starting from the highest priority mode (O/S), it is analyzed its corresponding conditions. If they are present, then the actual mode will be this one, otherwise it is necessary to check the conditions for the next lower priority mode (IMan, LO, Man, Auto, Cas, RCas and Rout) till the target mode, exclusive. For instance, if the target mode is Cas, it is necessary to check the conditions for O/S, IMan, LO, Man and Auto, in this order. If all those conditions are false, the actual mode will be the target mode.

Mode	Conditions
O/S	Resource block is in O/S (resource state is Standby) - Enumerated parameter has an invalid value
IMan	BKCAL_IN.status is Bad BKCAL_IN.status is Good – Fault State Active, Not Invited or Initialization Request.
LO	Fault state is active (in an output function block) CONTROL_OPTS.Track Enable active and TRK_IN_D is active. If target is Man, then the CONTROL_OPTS.Track in Manual must be active.
Man	Target mode has just changed from O/S

Mode	Conditions
	(Status attribute of primary input parameter (IN parameter) is Bad or Uncertain with option to treat Uncertain as Bad) and (Bypass not set). Target mode is RCas or ROut, and SHED_OPT=shed to Manual or shed to next
Auto	Target mode is Cas and (CAS_IN.status=Bad or cascade initialization not completed) Target mode is RCas and RCAS_IN.status=Bad and SHED_OPT=shed to Auto or shed to next Target mode is ROut and ROUT_IN.status=Bad and SHED_OPT = shed to Auto or shed to next.
Cas	Actual mode last execution was Cas. Target mode is Cas and cascade initialization has just completed Target mode is RCas and RCAS_IN.status=Bad and SHED_OPT=shed to next and cascade initialization has just completed Target mode is ROut and ROUT_IN.status=Bad and SHED_OPT=shed to next and cascade initialization has just completed
RCas	RCas cascade initialization has just completed or actual mode last execution was RCas.
ROut	ROut cascade initialization has just completed or actual mode last execution was ROut.

When the actual mode is different from the target, a good suggestion to find the cause is an analysis of the corresponding conditions for the actual mode. The most frequent causes are shown below.

Mode	Frequent cause
O/S	Check the Resource block mode and all enumerated parameters
Iman	Check the BKCAL_IN.status: Bad – No Comm : link failure in backward path (BKCAL_OUT -> BKCAL_IN) NI : lower block can not execute in Cas (check the target mode of lower block as well fault state condition) or link failure in forward path (OUT -> CAS_IN)
LO	Check : Value and status of TRK_IN_D, TRK_VAL and CONTROL_OPTS (Track Enable, Track in Manual) Output block : status of CAS_IN and the delay time for fault state established by FSTATE_TIME
Man	Status of IN If the target is Cas, check the links with the upper block in backward path (BKCAL_OUT -> BKCAL_IN) and forward path (OUT -> CAS_IN) If the block is not reaching the target mode Rcas or Rout, compare the updating rate of RCAS_IN and ROUT_IN by a Control Application to SHED_RCAS and SHED_ROUT.
Auto	If the target is Cas, check the links with the upper block in backward path (BKCAL_OUT -> BKCAL_IN) and forward path (OUT -> CAS_IN) If the block is not reaching the target mode Rcas or Rout, compare the updating rate of RCAS_IN and ROUT_IN by a Control Application to SHED_RCAS and SHED_ROUT.
Cas	If the block is not reaching the target mode Rcas or Rout, compare the updating rate of RCAS_IN and ROUT_IN by a Control Application to SHED_RCAS and SHED_ROUT.

f) Cascade initialization

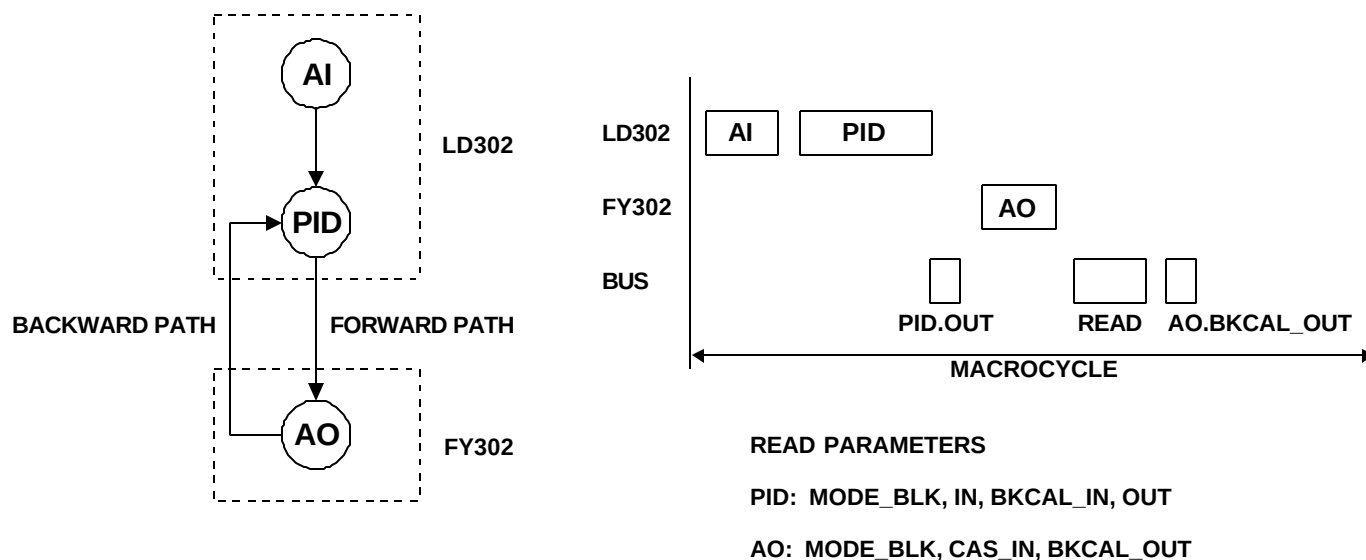


Fig.1 – Example to analyze the cascade initialization

The following table shows the sequence of status exchange between the PID and AO blocks for cascade initialization, and a link failure between PID.OUT and AO.CAS_IN (forward path) after execution 8.

PID parameters

Execution	1	2	3	4	5	6	7	8	9	10	11
Target	O/S	Auto									
IN	Bad	GNC	GNC	GNC	GNC	GNC	GNC	GNC	GNC	GNC	GNC
BKCAL_IN	Bad	Bad-O/S	NI	IR	GC	GC	GC	GC	NI	NI	NI
Actual	O/S	IMan	IMan	IMan	Auto	Auto	Auto	Auto	IMan	IMan	IMan
OUT	Bad-O/S	GC	GC	IA	GC	GC	GC	GC	GC	GC	GC

AO parameters

Execution	1	2	3	4	5	6	7	8	9	10	11
Target	O/S	Cas									
CAS_IN	Bad	GC	GC	IA	GC	GC	GC	Bad	Bad	Bad	Bad
Actual	O/S	Man	Auto	Cas	Cas	Cas	Cas	Auto	Auto	Auto	Auto
BKCAL_OUT	Bad-O/S	NI	IR	GC	GC	GC	GC	NI	NI	NI	NI

Meaning of status :

GNC – Good Non-Cascade – Non Specific

GC - Good Cascade – Non Specific

IA - Good Cascade – Initialization Acknowledge

IR - Good Cascade – Initialization Request

NI - Good Cascade – Not Invited

Bad – Bad – any sub-status

The above sequence for cascade initialization applies not only for the Cas mode, but also for Rcas and Rout modes.

g) Example

Mode configuration for the loop control in the figure 1.

	Supported	Target	Actual	Permitted	Normal
AI	O/S,Man,Auto	Auto	-	O/S,Auto	Auto
PID	O/S,Iman,LO,Man,Auto,Cas,Rcas,Rout	Auto	-	O/S,Man,Auto	Auto
AO	O/S,Iman,LO,Man,Auto,Cas,Rcas	Cas	-	O/S,Man,Auto,Cas	Cas

h) Specific information for driver developers

Internally, the each mode attribute is assigned within the bitstring in the following manner:

	Hex value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
O/S	0x80	1	0	0	X	X	X	X	X
Iman & LO	-	Not Valid Target Modes							
Man	0x10	0	0	0	1	0	X	X	X
Auto	0x08	0	0	0	0	1	0	0	0
Cas	0x0c	0	0	0	0	1	1	0	0
Rcas	0x0a	0	0	0	0	1	X	1	0
Rout	0x09	0	0	0	0	1	X	0	1

Where: X = indicates the bit setting is retained from the previous target mode.

Target and Normal Mode Bitstring

	Hex value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
O/S	0x80	1	0	0	0	0	0	0	0
Iman	0x40	0	1	0	0	0	0	0	0
LO	0x20	0	0	1	0	0	0	0	0
Man	0x10	0	0	0	1	0	0	0	0
Auto	0x08	0	0	0	0	1	0	0	0
Cas	0x04	0	0	0	0	0	1	0	0
Rcas	0x02	0	0	0	0	0	0	1	0
Rout	0x01	0	0	0	0	0	0	0	1

Actual Mode Bitstring

Block	Permitted modes	Hex value	O/S	-	-	Man	Auto	Cas	Rcas	Rout
AI	O/S,Auto	0x88	1	0	0	0	1	0	0	0
PID	O/S,Man,Auto	0x98	1	0	0	1	1	0	0	0
AO	O/S,Man,Auto,Cas	0x9c	1	0	0	1	1	1	0	0

Permitted Mode Bitstring

(Example for the loop control in the figure xx)

The retained target mechanism is suitable to work with toggle switch in the interface device following the rules:

A/M toggle switch:

“automatic mode” -> Man mode : Bit #4 <- 1 and Bit #3 <- 0

Man mode -> “automatic mode” : Bit #4 <- 0 and Bit #3 <- 1

Cascade/Local toggle switch:

Cascade -> Local : Bit #2 <- 0

Local >- Cascade : Bit #2 <- 1

Remote/Non-remote setpoint toggle switch:

Remote -> Non-remote : Bit #1 <- 0 and Bit #0 <- 0

Non-remote -> Remote : Bit #1 <-1 and Bit #0 <- 0

Remote/Non-remote output toggle switch:

Remote -> Non-remote : Bit #1 <- 0 and Bit #0 <- 0

Non-remote -> Remote : Bit #1 <- 0 and Bit #0 <- 1

Target mode	Rule for toggle	Value retained	Possible retained modes	Target + Retained modes
O/S : 100xxxxx	Bit #7 <- 0	000xxxxx	00010000 (0x10) : Man 00001000 (0x08) : Auto 00001100 (0x0c) : Cas 00001010 (0x0a) : Rcas 00001001 (0x09) : Rout	10010000 (0x90) : Man 10001000 (0x88) : Auto 10001100 (0x8c) : Cas 10001010 (0x8a) : Rcas 10001001 (0x89) : Rout
Man: 00010xxx	Bit #4 <- 0 and Bit #3 <- 1	00001xxx	00001000 (0x08) : Auto 00001100 (0x0c) : Cas 00001010 (0x0a) : Rcas 00001001 (0x09) : Rout	00010000 (0x10) : Auto 00010100 (0x14) : Cas 00010010 (0x12) : Rcas 00010001 (0x11) : Rout
Rcas: 00001x10	Bit #1 <- 0 and Bit #0 <- 0	00001x00	00001000 (0x08) : Auto 00001100 (0x0c) : Cas	00001010 (0x0a) : Auto 00001110 (0x0e) : Cas
Rout: 00001x01	Bit #1 <- 0 and Bit #0 <- 0	00001x00	00001000 (0x08) : Auto 00001100 (0x0c) : Cas	00001001 (0x09) : Auto 00001101 (0x0d) : Cas

Scaling Parameters

The scaling parameter defines the operating range and the engineering units associated with a parameter. It also defines the number of digits to the right of the decimal point, which should be used by an interface device in displaying that parameter.

Scaling information is used for two purposes. Display devices need to know the range for bar graphs and trending, as well as the units code. Control blocks need to know the range to use internally as percent of span, so that the tuning constants may remain dimensionless.

The PID blocks take the error and convert it to percent of span using the PV_SCALE. The algorithm operates on percent of span to produce an output in that form. This is converted back to a number with engineering units by using the range of OUT_SCALE.

The AI block has the XD_SCALE parameter to define the engineering units expected from the transducer.

The AO block uses the XD_SCALE to convert the SP value to the engineering unit expected by the output transducer block, that is also the engineering units of the readback value.

The following fields form the scale:

- Engineering Units at 100% of scale - The value that represents the upper end of range in engineering unit.
- Engineering Units at 0% of scale - The value that represents the lower end of range in engineering unit.
- Units Index - Device Description units code index for the engineering unit.
- Decimal Point - The number of digits to the right of the decimal point which should be used by an interface device in displaying the specified parameter.

Example Using Scale Parameter

The PID algorithm works internally with values in percent of span. Therefore the PID block converts the error to percentage (PV_SCALE), it calculates the output in percentage, and then it converts to engineering unit of output (OUT_SCALE).

1. The PID takes the input IN and SP and converts to percentage of the PV_SCALE :

$$\text{VALUE\%} = (\text{VALUE} - \text{EU}_0) * 100 / (\text{EU}_{100} - \text{EU}_0) \quad [\text{PV\_SCALE}]$$

PV_SCALE:

EU at 100% = 20

EU at 0% = 4

Units Index = mA

Decimal point = 2

SP = 15 mA

PV = 10 mA

The values of SP and PV in percentage are:

$$\text{SP\%} = (15 - 4) * 100 / (20 - 4) = 68.75\%$$

$$\text{PV\%} = (10 - 4) * 100 / (20 - 4) = 37.50\%$$

2. The PID algorithm calculates the error in percentage. If it is configured the reverse action, the error is the difference between SP% and PV%.

$$\text{Error\%} = \text{SP\%} - \text{PV\%} = 31.25\%$$

3. The PID algorithm applies the Error% to the calculation of the P, I and D terms. If only the proportional term is enabled, the value of the output is :

GAIN = 1.0

RESET = +INF

RATE = 0.0

OUT% = 31.25%

4. The output value is converted from percentage to engineering units of the OUT_SCALE:

$OUT = OUT\% / 100 * (EU_{100} - EU_0) + EU_0 \quad [OUT\_SCALE]$

OUT_SCALE:

EU at 100% = 15

EU at 0% = 3

Units Index = psi

Decimal point = 2

The output value of this example is:

$OUT = 31.25 / 100 * (15 - 3) + 3 = 6.75 \text{ psi}$

Fault State Handling

A) Definition

The Fault State is a special state that allows the output block to do safe action when it has been detected an abnormal situation or the user set to Fault State in the Resource block.

The abnormal situation occurs when there is an unusable input (bad sensor, for example) or the loss of the communication between function blocks longer than a specified time (FSTATE_TIME).

The blocks that support cascade control (as PID, OSDL and SPLT) propagate the fault state status forward to the output block.

When the condition that activated the Fault State is normalized, the Fault State is cleared and the block returns to the normal operation.

B) Generate the Initiate Fault State status (The fault was detected by the own block)

Blocks like PID, OSDL and SPLT may be configured to send an Initiate Fault State (IFS) status when they detect an unusable input. The bit "IFS if bad IN" and/or the bit "IFS if bad CAS_IN" in the STATUS_OPTS or OSDL_OPTS parameters must be true to generate an IFS status when the corresponding input is unusable.

C) Propagate the Initiate Fault State status (The fault occurred in the upstream block)

The blocks that support the cascade control have special handling to propagate the fault to the downstream blocks till the output block.

When the block, that is in cascade mode (Cas, RCas), receives an Initiate Fault State (IFS) status, then this status will be reported to forward path. For example, considering a PID block that is receiving a "Good Cascade IFS" status in CAS_IN input. If the target mode of the PID is Cas, then the IFS status will be the status of OUT, replacing the normal status. Otherwise, the IFS status will not be propagated forward.

D) Using the Resource Block to activate the Fault State

The Fault State of the Resource block forces all output function blocks into the device to go immediately to the fault state condition. The Resource block has some parameters that define the fault state behavior:

FEATURES_SEL – The "Fault State supported" bit is used to enable the Fault State characteristic into the Resource block. The default value is disabled.

FAULT_STATE – It only indicates the Fault State in the Resource block, not in individual output block. For example, if an AO block is in Fault State because its CAS_IN input is bad, the FAULT_STATE parameter will not be active.

SET_FSTATE – The user may force FAULT_STATE to active by setting this parameter to On.

CLEAR_FSTATE – The user may force FAULT_STATE to clear by setting this parameter to On.

Fault State Active

When the Output Function Blocks detects an abnormal condition, the block goes to a fault state. The abnormal conditions are:

Loss of communication to CAS_IN for a time that exceeds FSTATE_TIME;

IFS status in the CAS_IN input when the target mode is Cas, for a time that exceeds FSTATE_TIME;

IFS status in the RCAS_IN when the target mode is Rcas, for a time that exceeds FSTATE_TIME;

The FAULT_STATE parameter of the Resource block is Active because the user set to on the SET_FSTATE parameter and the bit "Fault State supported" in the FEATURES_SEL is true.

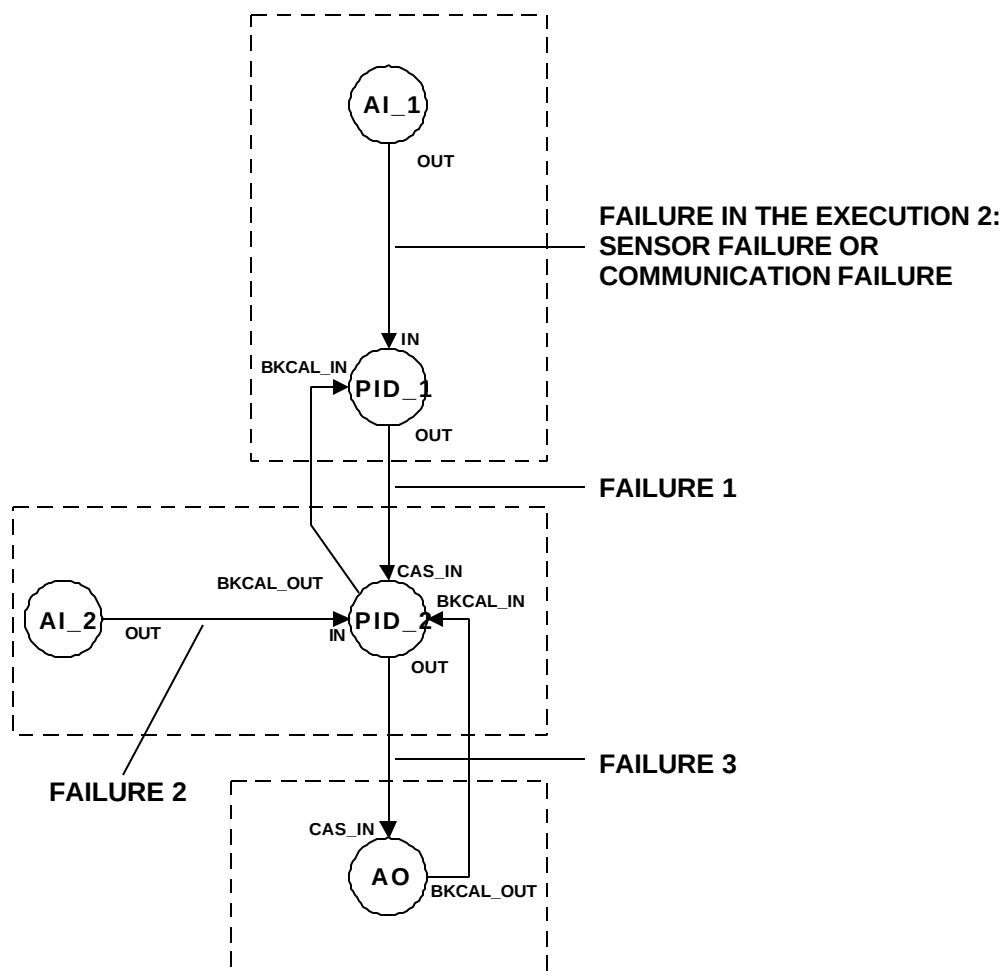
When the output block is in the Fault State the output may retain the last value or goes to the preset value determined by the FSTATE_VAL. The default is retaining the last value. The output goes to the preset value if the bit "Fault State to value" in the IO_OPTS is true.

When the Fault State is active then the actual mode of the output block goes to Local Override (LO). In the backward path, the block sends the Not Invited (NI) status to indicate that the block is in "Fault State".

Optionally, the target mode of the output block will be changed to Manual by the block algorithm when the Fault State is active. In order to set this characteristic the bit "Target to Man if Fault State activated" in IO_OPTS needs to be true.

Examples

Considering the following control loop using for the examples below. The below tables show the sequence of status exchange between the PID_1



Master, PID_2 Slave and AO blocks in a fault and normal condition.

The status of **PID_1.IN** becomes bad in the execution 2, which may be due to a Bad-Sensor Failure detected by the **AI_1** or a communication failure between **AI_1.OUT** and **PID_1.IN**. The **PID_2** Slave only propagates the status forward. The **AO** block receives the IFS status and active the Fault State. In the safety condition the preset value of the **FSTATE_VAL** is used to the output of the **AO** block. After the execution 6, the bad status in the **IN** of the PID master is cleared and the loop returns to normal operation.

PID 1 – Master

STATUS_OPTS = "IFS if Bad IN"

Execution	1	2	...	4	5	6	7	8	9	10	11
Target	Auto										
IN	GNC	Bad	...	Bad	Bad	Bad	Bad	GNC	GNC	GNC	GNC
BKCAL_IN	GC	GC	...	GC	GC	NI	NI	NI	NI	NI	IR
Actual	Auto	Man	...	Man	Man	Iman	Iman	Iman	Iman	Iman	Iman
OUT	GC	IFS	...	IFS	IFS	IFS	IFS	GC	GC	GC	IA

PID 2 – Slave

Execution	1	2	...	4	5	6	7	8	9	10	11
Target	Cas										
CAS_IN	GC	IFS	...	IFS	IFS	IFS	GC	GC	GC	IA	GC
BKCAL_IN	GC	GC	...	GC	NI	NI	NI	IR	GC	GC	GC
Actual	Cas	Cas	...	Cas	Iman	Iman	Iman	Iman	Auto	Cas	Cas
OUT	GC	IFS	...	IFS	IFS 100	IFS 100	GC 100	IA 100	GC	GC	GC
BKCAL_OUT	GC	GC	...	GC	NI	NI	NI	NI	IR	GC	GC

AO

FSTATE_VAL = 100

FSTATE_TIME = 2 sec

IO_OPTS = "Fault State to value"

PV_SCALE(E0%-E100%)=0-100

XD_SCALE (E0%-E100%)= 4-20

2 sec
|-----|

Execution	1	2	3	4	5	6	7	8	9	10	11
Target	Cas										
CAS_IN	GC	IFS	IFS	IFS	IFS 100	IFS 100	IFS 100	GC 100	IA 100	GC	GC
SP	GC	IFS	IFS	IFS 100	IFS 100	IFS 100	IFS 100	GC 100	IA 100	GC	GC
Actual	Cas	Cas	Cas	LO	LO	LO	LO	Auto	Cas	Cas	Cas
OUT	GC	GC	GC	GC 20	GC 20	GC 20	GC 20	GC 20	GC 20	GC	GC
BKCAL_OUT	GC	GC	GC	NI	NI	NI	NI	IR	GC	GC	GC

Other link failures in the forward path could force the AO into Fault State since the configuration is as it follows:

Failure 1) **PID_1.OUT to PID_2.CAS_IN**: PID_2.STATUS_OPTS = "IFS if Bad CAS_IN"

Failure 2) **AI_2.OUT to PID_2.IN**: PID_2.STATUS_OPTS = "IFS if Bad IN"

Failure 3) **PID_2.OUT to AO.CAS_IN**: No configuration is required for the AO block to force it to Fault State.

Note:

FSTATE_VAL has the same engineering unit of SP. Therefore it is using the PV_SCALE, not the XD_SCALE.

When the Resource block forces all output blocks to Fault State, the FSTATE_TIME is not used.

Alarms and Events – Alert Processing

Alarms and events, known as alerts, represent state changes within function block applications. In detection of a significant event, a function block may report this event to an interface device or other field devices.

Alarms refer not only to the comparison between a variable and a limit, but also what is called block alarm, that is used to report errors found in the software and/or hardware during block execution.

For alarms, both entering and exiting alarm condition are defined as an alert state, which will cause a report message to be published onto the network. The time at which the alert state was detected is included as a time stamp in the alert message. The reporting of alerts may be individually suppressed by setting the corresponding priority.

Update event is a mechanism used to inform an interface device that a static parameter was changed so the parameter is read only in this case. It is a very optimized way to keep track of such kind of parameters without doing polling, because these parameters are changed very rarely compared with dynamic parameters.

a) Alarm parameter (X_ALM parameter)

The alarm parameter is provided in a block to capture the dynamic information associated with an alarm. The information contained in the alarm parameter is transferred to an alert object when the alarm is reported. The following fields form the alarm parameter:

Unacknowledged

Alarm state

Time stamp

Subcode

Value

Each one of these fields is explained below.

Unacknowledged

When it is detected a positive transition of alarm state (entering in alarm active), it will be set to Unacknowledged. This field is set to Acknowledged

when the plant operator acknowledges the alarm through the interface device, which is responsible for alarm management.

It is possible to configure the auto-acknowledgement feature for each type of alarm for the block through the ACK_OPTION parameter. If a positive transition of alarm type happens and the corresponding bit in the ACK_OPTION is true, then it will not be required an operator plant acknowledgement.

The other form to auto-acknowledge the alarm notice is configuring the alert-priority of the respective alarm to 0,1 or 2. The alert-priority will be discussed later.

Unacknowledged will have the following enumerations:

- 0 = Undefined
- 1 = Acknowledged
- 2 = Unacknowledged

Alarm state

This field gives an indication of whether the alert is active and whether it has been reported. The Alarm State will have the following bit enumeration:

- 1 – Clear-Reported
- 2 – Clear-Not Reported
- 3 – Active-Reported
- 4 – Active-Not Reported

The alarm state is cleared when the block goes to Out of service mode.

Time stamp

Time stamp is the time when the change in alarm state was detected that is unreported. This value will be maintained constant until alert confirmation has been received.

Sub code

This field contain an enumeration specifying the cause of the alert to be reported.

Value

The value of the associated parameter at the time the alert was detected.

b) Alarm limit (X_LIM parameter)

An analog alarm occurs when a value meets or exceeds a limit. For a high alarm, an alarm is true when the analog value is greater than the limit. The status of the alarm remains true until the value drops below the limit minus the alarm hysteresis.

The alarm type can be disabled setting its respective alarm limit parameter to +/- infinity, which is the default of all alarm limits.

The analog parameter compared to alarm limit depends on the block type:

PID: PV and (PV-SP), regardless of CONTROL_OPTS.Direct acting

Analog alarm: PV

AI: OUT

Setpoint generator: BKCAL_IN – OUT

c) Alarm hysteresis (ALARM_HYS parameter)

Amount the PV or OUT must return within the alarm limits before the alarm condition clears. Alarm Hysteresis is expressed as a percent of the PV/OUT span.

The span used depends on the block type:

PID: PV_SCALE

AI, Setpoint generator, Analog alarm: OUT_SCALE

d) Alert Priority (X_PRI parameter)

The alert priority is a parameter, which may be used to configure the priority value assigned to an alarm or event.

The Alert Priority can be:

0-1 – The associated alert is not sent as a notification. The priority is above 1, then the alert must be reported. This priority is auto-acknowledged.

2 – Reserved for alerts that do not require the attention of a plant operator. Block alarm and update event have this priority. This priority is auto-acknowledged.

3-7 – Advisory Alarms. In this priority is necessary to sent an acknowledged.

8-15 – Critical Alarms. In this priority is necessary to sent an acknowledged.

e) Alert key (ALERT_KEY parameter)

It is an identification number of the plant unit. This information may be used in the host for sorting alarms, etc.

f) Alarm Summary (ALM_SUM parameter)

The parameter Alarm Summary summarizes the status of up to 16 process alarms of the same block. For each alarm, the current status, unacknowledged status, unreported status, and disabled status are maintained.

It has four attributes:

Current Alarms - the Active status of each alarm.

Unacknowledged - the Unacknowledged status of each alarm.

Unreported - the Unreported status of each alarm.

Disabled - the Disabled status of each alarm.

g) ACK_OPTION parameter

Selection of whether alarms associated with the block will be automatically acknowledged.

Bit = 0 -> auto-acknowledgment disabled

Bit = 1 -> auto-acknowledgment enabled

The correspondent bits for each type of alarm in the ACK_OPTION is the same of that defined for ALARM_SUM except for the resource block.

Bit	Meaning
0	Discrete alarm
1	High High alarm
2	High alarm
3	Low Low alarm
4	Low alarm
5	Deviation High alarm
6	Deviation Low alarm
7	Block alarm
8	Not used
9	Not used
10	Not used
11	Not used
12	Not used
13	Not used
14	Not used
15	Not used

Association of bit number to alarm type

(ALARM_SUM and ACK_OPTION)

The ACK_OPTION of the resource block has a different association of bit number to alarm type from the previous table, its special meaning is described below:

Bit #1 – Writes have been enabled (WRITE_ALM)

Bit #8 – Block Alarm (BLOCK_ALM)

h) FEATURES_SEL parameter

It is a resource block parameter that has an element to enable/disable alert report for the whole resource, "report supported".

i) CONFIRM_TIME parameter

A reply is required that confirms receipt of the alert notification. If the reply is not received within a time-out period (CONFIRM_TIME), the alert will be re-transmitted. It is a resource block parameter, so it is valid for all alert of that resource.

Therefore the alarm parameter is a structured object, which is defined in conjunction with other parameters:

Enable / disable alarm evaluation :

ALARM_SUM : enable/disable each alarm type for a specific block

X_LIM : it is possible to disable the alarm evaluation by setting the limit to +INF or -INF

Enable / disable alarm report :

FEATURES_SEL: setting the bit "report supported" enables alarm report for the whole resource

X_PRI: besides configuring the FEATURES_SEL, it is necessary to set the priority of alarm greater than or equal to 2 to enable the alarm report

Auto-acknowledgment :

X_PRI: the corresponding alarm will be auto-acknowledged if the alarm priority is 0,1 or 2

ACK_OPTION: this parameter makes possible to enable/disable auto-acknowledgment for each alarm type, regardless the configuration of X_PRI

The information contained in the alarm parameter is transferred to an alert object when the alarm is reported (if it is enabled) to an interface device. As an example, the following parameters are used to configure the Low Low Alarm of the PID block: LO_LO_PRI (alarm priority), LO_LO_LIM (limit parameter) and LO_LO_ALM (alarm parameter), ALARM_SUM, ACK_OPTION.

Summarizing, the function block detects the alarm condition. The communication stack is responsible to send the alert report to the interface device, which must reply to device, otherwise the alert report will be re-transmitted after a timeout defined by CONFIRM_TIME. The alarm configuration (ACK_OPTION and priority) may require that a plant operator acknowledge it even though the condition has cleared.

j) Example of analog alarm

It follows an example of AI block alarm configuration, that illustrates how is the alarm processing as well the corresponding alert report.

RS parameter:

FEATURES_SEL = Reports supported

CONFIRM_TIME = 640 000 (20 seconds, multiple of 1/32 milliseconds)

AI parameters:

ALERT_KEY = 12 (this value could be related to a boiler, for example, therefore any alert received by interface device with this alert key means an alert in that boiler)

OUT_SCALE.EU at 100% = 200

OUT_SCALE.EU at 0% = 0

HI_LIM = 190

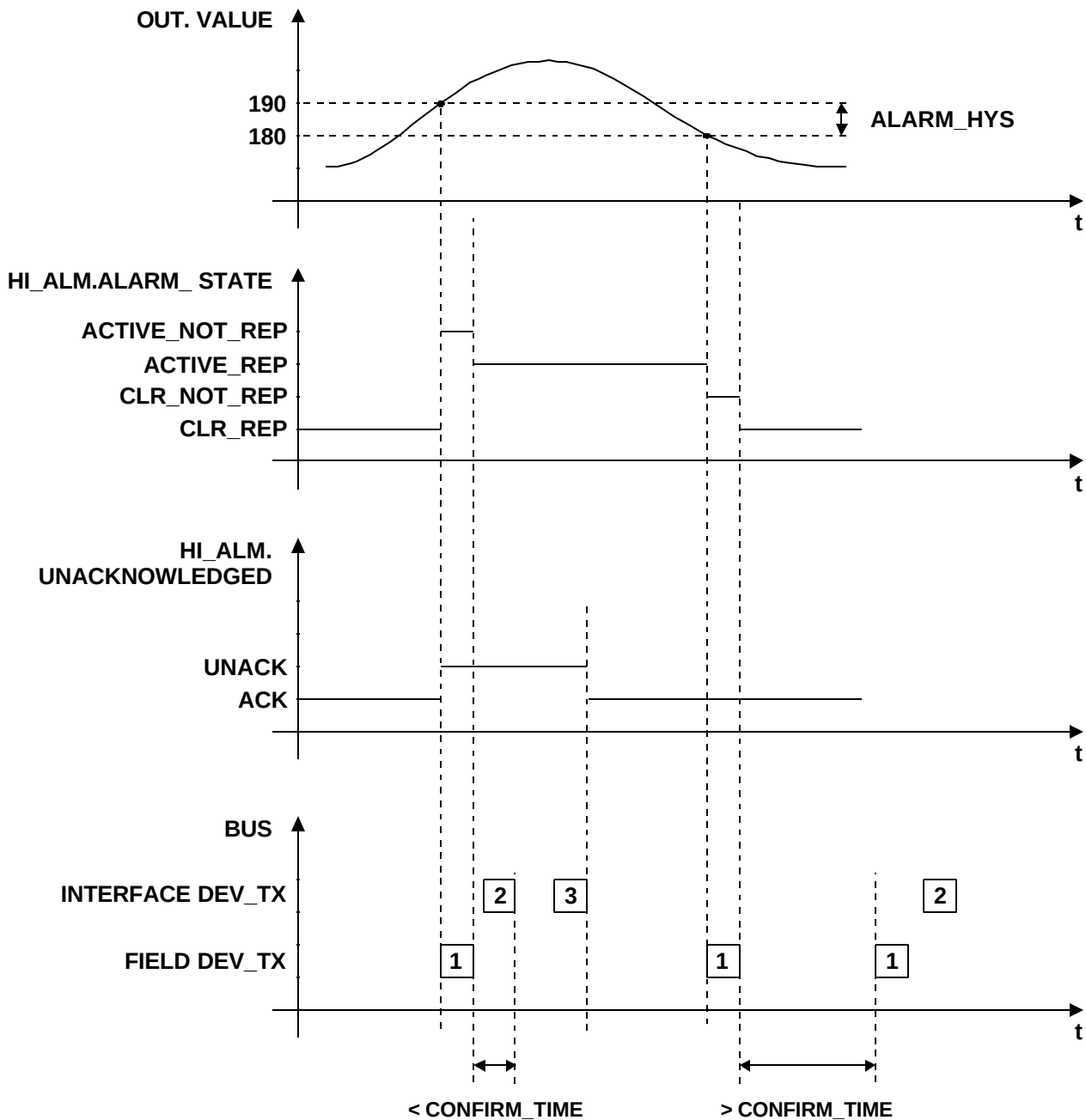
HI_PRI = 4

ALARM_HYS = 5%

ALARM_SUM.DISABLED = Discrete, HiHi, LoLo, Lo, DevHi, DevLo, BlockAlm

ACK_OPTION = 0x00

Only the high alarm is enabled in ALARM_SUM.DISABLED and it is disabled the auto-acknowledgement (HI_PRI=4 and bit reset in ACK_OPTION), therefore it is necessary an operator plant intervention.



MESSAGE 1: ALERT NOTIFY

MESSAGE 2: ALERT NOTIFY - CONFIRMATION

MESSAGE 3: ALERT ACK (OPERATOR PLANT INTERVENTION)

Fig. 1 – Alert processing

k) Block alarm (BLOCK_ALM parameter)

The block alarm is used for all configuration, hardware, connection failure or system problems in the block. These problems detected by the block algorithm are registered in the BLOCK_ERR parameter, which is a bit string, so that multiple errors may be shown.

Block error conditions is defined (0= inactive, 1 = active) as follows:

- 0 = Other (LSB)
- 1 = Block Configuration Error
- 2 = Link Configuration Error
- 3 = Simulate Active
- 4 = Local Override
- 5 = Device Fail Safe Set
- 6 = Device Needs Maintenance Soon
- 7 = Input Failure/ process variable has BAD status
- 8 = Output Failure
- 9 = Memory Failure
- 10 = Lost Static Data
- 11 = Lost NV Data
- 12 = Readback Check Failed
- 13 = Device Needs Maintenance Now
- 14 = Power-up
- 15 = Out-of-Service (MSB)

The cause of the alert is entered in the Subcode field of BLOCK_ALM, for example, value 3 means Simulate Active. The first condition to become active will set to Active the Alarm State attribute, if other conditions also become active, no alert will be sent. When all conditions become inactive, then an alert with Clear will be reported.

Block alarm has a fixed priority of 2, therefore it is auto-acknowledged (no operator plant intervention is required).

l) Update Event (UPDATE_EVT parameter)

The update event parameter is provided in a block to capture the dynamic information associated with a write to a static parameter within the block. An update alert object transfers the information contained in the update event parameter when the alert is reported.

The index of the changed parameter (relative to the start of the function block in the OD) and the new static revision level (ST_REV) are also included in the alert message.

Update event has a fixed priority of 2, therefore it is auto-acknowledged (no operator plant intervention is required).

Simulation

All input and output class function blocks have a SIMULATE or SIMULATE_D or SIMULATE_P parameter, which has a pair of status and values, and an enable switch. This parameter acts as a switch at the interface between the I/O function block and the associated transducer block or hardware channel.

Enable simulation

The simulate jumper must be ON to enable simulation in the SIMULATE (SIMULATE_D or SIMULATE_P) parameter.

The BLOCK_ALM and BLOCK_ERR parameters will show the simulation condition (enable or disable). Such parameters in the Resource Block (RS) will indicate the condition of simulate jumper, while in the Input/Output Function Blocks they will indicate the enable switch condition in the SIMULATE (SIMULATE_D or SIMULATE_P) parameter.

Simulation disabled

When disabled, the SIMULATE.Simulate Value and Status will track SIMULATE.Transducer Value and Status in order to provide a bumpless transfer from disabled to enabled. The parameter will always initialize to disabled and will be stored in dynamic memory.

Input Function Block (AI,DI,PUL)

The SIMULATE.Transducer Status and SIMULATE.Transducer Value come from the transducer block or input channel, and contain what will be sent to the input block if the switch is off (disabled). The SIMULATE.Simulate Value and SIMULATE.Simulate Status are presented to the input block when the enable switch is on, and the transducer block or input channel is ignored. The status can be used to simulate transducer faults. The transducer value and status will always be written with transducer data at each evaluation of the input function block.

Simulation condition	Action
Enable	SIMULATE.Simulate Value and Status -> PV (after scaling,linearization and filtering)
Disable	SIMULATE.Transducer Value and Status -> PV (after scaling,linearization and filtering) and SIMULATE.Simulate Value and Status

Output Function Block (AO,DO)

The SIMULATE.Simulate Value and SIMULATE.Simulate status become the READBACK value and status when the enable switch is on, and the transducer block is ignored. The status can be used to simulate transducer faults. The transducer attribute value and status reflect the transducer readback value and status when simulation is enabled and the transducer maintains last output and ignores the OUT of the Output block.

Simulation condition	Action
Enable	SIMULATE.Simulate Value and Status -> READBACK
Disable	SIMULATE.Transducer Value and Status -> READBACK and SIMULATE.Simulate Value and Status

CHANNEL configuration

The CHANNEL parameter configuration depends on the device features as it follows:

a) Fixed I/O device : This type of device has a fixed number of I/O. All Smar field devices belong to this class.

The channel is numbered from 0 to the maximum number of I/O.

b) Configurable I/O device : The user may configure the number of I/O modules as well the I/O type (input or output ,discrete,analog,pulse,...). The DFI302 is the only device classified as a configurable I/O device.

All I/O modules have the I/O points arranged as it follows:

- Point (P) : Ordinal number of I/O point in a group, it is numbered from 0 (first point) to 7(last point in the group), and 9 means the whole group of points.

- Group (G) : Ordinal number of group in the specified I/O module, it is numbered from 0 (first group) till number of groups minus 1.

In the DFI302, the I/O modules are arranged following a hierarchy:

- Slot (S): One slot supports one I/O module, and it is numbered from 0 (first slot in the rack) till 3 (last slot in the rack).

- Rack (R): Each rack has four slots. The rack is numbered from 0 (first rack) till 14 (last rack). Therefore a single I/O point in the DFI302 may be identified by specifying the rack (R) , slot (S), group (G) and point (P). As the CHANNEL parameter in the multiple I/O blocks (MIO) must specify the whole group (8 points), the point must be 9, that means the whole group.

The value in the CHANNEL parameter is composed by that elements in the following form: RRS GP. For example, a CHANNEL parameter equals to 1203, it means rack 1, slot 2, group 0 and point 3. If CHANNEL parameter of a MAI block is 10119, it means rack 10, slot 1, group 1 and point 9 (whole group).

Before setting the CHANNEL parameter, it is recommended to configure the hardware in the HC block. Because the write check will verify if the I/O type configured in the HC block is suitable for block type. Therefore setting the CHANNEL parameter of AI block to access an I/O type different of analog input will be rejected.

Block Instantiation

Before explaining block instantiation it is better to clarify some concepts:

Block type : It is an algorithm to process the input parameters based on the configuration in the contained parameters, then it generates the outputs. It includes also method to write/read the parameters, DD and others. Indeed all this information are stored in Flash memory of device, therefore one device type has a predefined set of block type available in its firmware.

Block (block instance) : It is a block type associated to a database where the block parameters are stored (RAM and non-volatile memory).

Action Object : Through the action object, a block may be instantiated (created) or deleted. Before instantiating a block, it will be checked if the device supports the specified block type as well if there is available RAM and non-volatile memory to store the parameters.

All Smar devices support block instantiation and the Block Library (set of block type) for each type of device is shown in the item "Block type availability and initial block set".

Order of Parameters during Download

Some block parameters have a write check based on the value of others parameters. Such relationships are shown in the block parameter table of each block type in the columns "Valid Range" and "Store/Mode".

It follows the most common parameter relationship used in write check:

- It is required a mode to write the parameter.
- The valid range depends on a scale parameter
- For configurable I/O device, the CHANNEL parameter depends on the hardware configuration in the HC block.

Due to these relationships between parameters in the write check, some times it is necessary to take care about the order of parameters during a download of configuration.

The Smar configuration tool Syscon always sends the command to write in the MODE_BLK parameter as last one for that block, it is enough to avoid a lot of problems in the download. The user must observe the other cases and may change the parameter order easily using drag and drop in Syscon.

Data Type and Data Structure Definition

In this section are defined every data structure and data types used in the system.

Data Type	Description
Boolean	True or false
Integer8	
Integer16	
Integer32	
Unsigned8	
Unsigned16	
Unsigned32	
FloatingPoint	
VisibleString	they are one byte per character, and include the 7 bit ASCII character set.
OctetString	Octet strings are binary.
Date	
TimeOfDay	
TimeDifference	
BitString	
DateTimeValue	

Block Structure – DS-64

This data structure consists of the attributes of a block.

E	Element Name	Data Type	Size
1	Block Tag	VisibleString	32
2	DD MemberId	Unsigned32	4
3	DD ItemId	Unsigned32	4
4	DD Revision	Unsigned16	2
5	Profile	Unsigned16	2
6	Profile Revision	Unsigned16	2
7	Execution Time	Unsigned32	4
8	Period of Execution	Unsigned32	4
9	Number of Parameters	Unsigned16	2
10	Next FB to Execute	Unsigned16	2
11	Starting Index of Views	Unsigned16	2
12	NumberOfVIEW_3	Unsigned8	1
13	NumberOfVIEW_4	Unsigned8	1

Value & Status - Floating Point Structure – DS-65

This data structure consists of the value and status of floating point parameters that are Inputs or Outputs.

E	Element Name	Data Type	Size
1	Status	Unsigned8	1
2	Value	Float	4

Value & Status - Discrete Structure – DS-66

This data structure consists of the value and status of discrete value parameters.

E	Element Name	Data Type	Size
1	Status	Unsigned8	1
2	Value	Unsigned8	1

Scaling Structure – DS-68

This data structure consists of the static data used to scale floating point values for display purposes.

E	Element Name	Data Type	Size
1	EU at 100%	Float	4
2	EU at 100%	Float	4
3	Units Index	Unsigned16	2
4	Decimal Point	Integer8	1

Mode Structure – DS-69

This data structure consists of bit strings for target, actual, permitted, and normal modes.

E	Element Name	Data Type	Size
1	Target	Bitstring	1
2	Actual	Bitstring	1
3	Permitted	Bitstring	1
4	Normal	Bitstring	1

Access Permissions – DS-70

This data structure consists of access control flags for access to block parameters.

E	Element Name	Data Type	Size
1	Grant	Bit String	1
2	Deny	Bit String	1

Alarm Float Structure – DS-71

This data structure consists of data that describes floating point alarms.

E	Element Name	Data Type	Size
1	Unacknowledged	Unsigned8	1
2	Alarm State	Unsigned8	1
3	Time Stamp	Time Value	8
4	Subcode	Unsigned16	2
5	Value	Float	4

Alarm Discrete Structure – DS-72

This data structure consists of data that describes discrete alarms.

E	Element Name	Data Type	Size
1	Unacknowledged	Unsigned8	1
2	Alarm State	Unsigned8	1
3	Time Stamp	Time Value	8
4	Subcode	Unsigned16	2
5	Value	Unsigned8	1

Event Update Structure – DS-73

This data structure consists of data that describes a static revision alarm.

E	Element Name	Data Type	Size
1	Unacknowledged	Unsigned8	1
2	Update State	Unsigned8	1
3	Time Stamp	Time Value	8
4	Static Revision	Unsigned16	2
5	Relative Index	Unsigned16	2

Alarm Summary Structure – DS-74

This data structure consists of data that summarizes 16 alerts.

E	Element Name	Data Type	Size
1	Current	Bit String	2
2	Unacknowledged	Bit String	2
3	Unreported	Bit String	2
4	Disabled	Bit String	2

Simulate - Floating Point Structure – DS-82

This data structure consists of a simulate and transducer floating point value and status and a simulate enable/disable discrete.

E	Element Name	Data Type	Size
1	Simulate Status	Unsigned8	1
2	Simulate Value	Float	4
3	Transducer Status	Unsigned8	1
4	Transducer Value	Float	4
5	Simulate En/Disable	Unsigned8	1

Simulate - Discrete Structure – DS-83

This data structure consists of a simulate and transducer discrete value and status and a simulate enable/disable discrete.

E	Element Name	Data Type	Size
1	Simulate Status	Unsigned8	1
2	Simulate Value	Unsigned8	4
3	Transducer Status	Unsigned8	1
4	Transducer Value	Unsigned8	4
5	Simulate En/Disable	Unsigned8	1

Test Structure – DS-85

This data structure consists of function block test read/write data.

E	ElementName	DataType	Size
1	Value1	Boolean	1
2	Value2	Integer8	1
3	Value3	Integer16	2
4	Value4	Integer32	4
5	Value5	Unsigned8	1
6	Value6	Unsigned16	2
7	Value7	Unsigned32	4
8	Value8	FloatingPoint	4
9	Value9	VisibleString	32
10	Value10	OctetString	32
11	Value11	Date	7
12	Value12	Time of Day	6
13	Value13	Time Difference	6
14	Value14	Bitstring	2
15	Value15	Time Value	8

Manufacturer Specific Data Structure

In this section are defined manufacturer specific data structure used in the system.

Scaling Conversion Structure - DS-256

This data structure consists of data used to generate constants A and B in equation $Y = A * X + B$.

E	Element Name	Data Type	Size
1	From EU 100%	Float	4
2	From EU 0%	Float	4
3	To EU 100%	Float	4
4	To EU 0%	Float	4
5	Data Type	Unsigned8	1

Scaling Conversion Structure with Status - DS-257

This data structure consists of data used to generate constants A and B in equation $Y = A * X + B$ plus the output status.

E	Element Name	Data Type	Size
1	From EU 100%	Float	4
2	From EU 0%	Float	4
3	To EU 100%	Float	4
4	To EU 0%	Float	4
5	Data Type	Unsigned8	1
6	Output Status	Unsigned8	1

Scaling Locator Structure - DS-258

This data structure consists of data used to generate constants A and B in equation $Y = A * X + B$ plus the addresses in a slave device.

E	Element Name	Data Type	Size
1	From EU 100%	Float	4
2	From EU 0%	Float	4
3	To EU 100%	Float	4
4	To EU 0%	Float	4
5	Data Type	Unsigned8	1
6	Slave Address	Unsigned8	1
7	Modbus Address of Value	Unsigned16	2

Scaling Locator Structure with Status- DS-259

This data structure consists of data used to generate constants A and B in equation $Y = A \cdot X + B$ plus the addresses in a slave device.

E	Element Name	Data Type	Size
1	From EU 100%	Float	4
2	From EU 0%	Float	4
3	To EU 100%	Float	4
4	To EU 0%	Float	4
5	Data Type	Unsigned8	1
6	Slave Address	Unsigned8	1
7	Modbus Address of Value	Unsigned16	2
8	Modbus Address of Status	Unsigned16	2

Modbus Variable Locator Structure - DS-260

This data structure consists of data indicating the addresses in a slave device.

E	Element Name	Data Type	Size
1	Slave Address	Unsigned8	1
2	Modbus Address of Value	Unsigned16	2

Modbus Variable Locator Structure with Status- DS-261

This data structure consists of data indicating the addresses in a slave device.

E	Element Name	Data Type	Size
1	Slave Address	Unsigned8	1
2	Modbus Address of Value	Unsigned16	2
3	Modbus Address of Status	Unsigned16	2

FF Parameter ID Structure - DS-262

This data structure consists of data informing the position of the FF parameter requested.

E	Element Name	Data Type	Size
1	Block Tag	VisibleString(32)	32
2	Relative Index	Unsigned16	2
3	Sub Index	Unsigned8	1

Slave Address Structure - DS-263

This data structure consists of data informing the IP address and the Modbus address of the slaves.

E	Element Name	Data Type	Size
---	--------------	-----------	------

1	IP Slave1	VisibleString(16)	16
2	IP Slave2	VisibleString(16)	16
3	IP Slave3	VisibleString(16)	16
4	IP Slave4	VisibleString(16)	16
5	IP Slave5	VisibleString(16)	16
6	IP Slave6	VisibleString(16)	16
7	IP Slave7	VisibleString(16)	16
8	IP Slave8	VisibleString(16)	16
9	Slave Address1	Unsigned8	1
10	Slave Address2	Unsigned8	1
11	Slave Address3	Unsigned8	1
12	Slave Address4	Unsigned8	1
13	Slave Address5	Unsigned8	1
14	Slave Address6	Unsigned8	1
15	Slave Address7	Unsigned8	1
16	Slave Address8	Unsigned8	1

2 – Block Library

2.1 - Description of Block Types

RESOURCE	DESCRIPTION
RS	RESOURCE – This block contains data that is specific to the hardware that is associated with the resource.

TRANSDUCER BLOCKS	DESCRIPTION
DIAG	DIAGNOSTICS TRANSDUCER – It provides online measurement of block execution time, check of links between blocks and other features
DSP	DISPLAY TRANSDUCER – This block supported by devices with LCD display can be used to monitor and actuate in local parameters of blocks.
HC	HARDWARE CONFIGURATION TRANSDUCER – It configures the module type for each slot in the DFI302.
MBCF	MODBUS CONFIGURATION – This transducer block is used to configure general features related to Modbus gateway.

INPUT TRANSDUCER BLOCKS	DESCRIPTION
LD292/LD302	PRESSURE TRANSDUCER – This is the transducer block for LD292/LD302, a pressure transmitter.
TT302	TEMPERATURE TRANSDUCER – This is the transducer block for TT302, a temperature transmitter.
IF302	CURRENT FIELD BUS TRANSDUCER – This is the transducer block for IF302, a Current to Fieldbus Transmitter.
TP302	POSITION FIELD BUS TRANSDUCER – This is the transducer block for TP302, a Position Fieldbus Transmitter.
TEMP	DF-45 TEMPERATURE TRANSDUCER – This is the transducer block for the module DF-45, an eight low signal input module for RTD, TC, mV, Ohm.

OUTPUT TRANSDUCER BLOCKS	DESCRIPTION
FY302	FIELD BUS POSITIONER TRANSDUCER - This is the transducer block for FY302, a Fieldbus Positioner.
FP302	FIELD BUS PRESSURE TRANSDUCER - This is the transducer block for FP302, a Fieldbus to Pressure Converter.
FI302	FIELD BUS CURRENT TRANSDUCER - This is the transducer block for IF302, a Fieldbus to Current Converter.

INPUT FUNCTION BLOCKS	DESCRIPTION
AI	ANALOG INPUT – This block takes the analog input data from the analog input signal and it makes available to other function blocks. It has scaling conversion, filtering, square root, low cut and alarm processing.
DI	DISCRETE INPUT – This block takes the discrete input data from the discrete input signal, and it makes available to other function blocks. It has option to invert, filtering and alarm processing.
MAI	MULTIPLE ANALOG INPUT OUTPUT – It provides a way to receive 8 analog variables from other modules.
MDI	MULTIPLE DISCRETE INPUT OUTPUT – It provides a way to receive 8 discrete variables from other modules.
PUL	PULSE INPUT – It provides an analog value that represents a totalization of pulses in a physical discrete input.

CONTROL AND CALCULATION FUNCTION BLOCKS	DESCRIPTION
PID	PID CONTROL – This standard block has a lot of valuable features as setpoint treatment (value and rate limiting), filtering and alarm on PV, feedforward, output tracking and others.
EPID	ENHANCED PID – It has all the standard features plus : bumpless or hard transfer from a “manual” mode to an “automatic” mode and bias.
APID	ADVANCED PID – It has all the standard features plus: bumpless or hard transfer from a “manual” mode to an “automatic” mode, bias, adaptive gain, PI sampling, deadband for error, special treatment for error, ISA or parallel algorithm,..
ARTH	ARITHMETIC – This calculation block provides some pre-defined equations ready for use in applications as flow compensation, HTG, ratio control and others.
SPLT	SPLITTER – This block is used in two typical applications: split ranging and sequencing. It receives the output of PID block, that is processed according to the selected algorithm, then it generates the values for the two analog output blocks.
CHAR	SIGNAL CHARACTERIZER – It has capability for two signal characterization based on the same curve. The second input has an option for swapping “x” to “y”, providing an easy way to use the inverse function, that may be used in signal characterization of readback variables.
INTG	INTEGRATOR – It integrates a variable in function of the time. There is a second flow input that may be used for the following applications: net flow totalization, volume/mass variation in vessels and precise flow ratio control.
AALM	ANALOG ALARM – This alarm block has dynamic or static alarm limits, hysteresis, temporary expansion of alarm limits on step setpoint changes to avoid nuisance alarms, two levels of alarm limits and delay for alarm detection.
ISEL	INPUT SELECTOR – This block has four analog inputs that may be selected by an input parameter or according to a criterion as first good, maximum, minimum, middle and average.
SPG	SETPOINT RAMP GENERATOR – This block generates setpoint following a profile in function of the time. Typical applications are temperature control, batch reactors, etc.
TIME	TIMER and Logic – This block has four discrete inputs, that are processed by a combination logic. The selected timer processing type operates on the combined input signal to produce a measurement, delay, extension, pulse or debounce.
LLAG	LEAD-LAG – This block provides dynamic compensation of a variable. It is used normally in a feedforward control.
OSDL	OUTPUT SELECTOR / DYNAMIC LIMITER – It has two algorithms: Output selector – selection of output by a discrete input Dynamic limiter – this algorithm was developed specially for double cross limit in combustion control.
DENS	DENSITY – This block has a special algorithm to calculate the density in different types of engineering units: plato degree, INPM and others.
CT	CONSTANT – It provides analog and discrete output parameters with constant values.
FFET	FLIP-FLOP AND EDGE TRIGGER – It can be configured to work as SR flip-flop, RS flip-flop, D-LATCH and EDGE TRIGGER (rising, falling or bi-directional)
MBCS	MODBUS CONTROL SLAVE – When the device is working as gateway between Foundation Fieldbus and Modbus (slave device), this block may be used to exchange control data between both protocols.

CONTROL AND CALCULATION FUNCTION BLOCKS	DESCRIPTION
MBSS	MODBUS SUPERVISION SLAVE – When the device is working as gateway between Foundation Fieldbus and Modbus (slave device), this block may be used to convert Foundation Fieldbus parameters into Modbus variables. Such variables will be available to the supervisory with a Modbus driver.
MBCM	MODBUS CONTROL MASTER – When the device is working as gateway between Foundation Fieldbus and Modbus (master device), this block may be used to exchange control data between both protocols.
MBSM	MODBUS SUPERVISION MASTER – When the device is working as gateway between Foundation Fieldbus and Modbus (master device), this block may be used to convert Modbus variables into Foundation Fieldbus parameters. Such parameters will be available to the supervisory with a Foundation Fieldbus driver (OPC).

OUTPUT FUNCTION BLOCKS	DESCRIPTION
AO	ANALOG OUTPUT – The AO block provides an analog value to generate an analog output signal. It provides value and rate limiting, scaling conversion, fault state mechanism and other features.
DO	DISCRETE OUTPUT – The DO block provides a discrete value to generate a discrete output signal. There is option to invert the discrete value, fault state mechanism and other features.
MAO	MULTIPLE ANALOG OUTPUT – It provides a way to send 8 analog variables to other modules.
MDO	MULTIPLE DISCRETE OUTPUT OUTPUT – It provides a way to send 8 discrete variables to other modules.
STEP	STEP OUTPUT PID – It is used when the final control element has an actuator driven by an electric motor.

2.2 - Block type availability and initial block set

The table below shows how powerful and flexible the Smar devices are. For example, the user may instantiate up to 20 blocks selected from 17 block types (algorithms) in a field device as LD302. Indeed it means that almost all control strategy may be implemented using only the Smar field devices.

Read carefully the notes in order to fully understand the information in this table.

Block type	LD292/ LD302	TT302	IF302	TP302	FY302	FP302	FI302	FB700	DFI302
RS (1)	1	1	1	1	1	1	1	1	1
DIAG (1)	1	1	1	1	1	1	1	1	1
DSP (1)	1	1	1	1	1	1	1		
HC (1)									1
MBCF(1)									0
TRD-LD (1)	1								
TRD-TT (2)		2							
TRD-IF (3)			3						
TRD-TP(1)				1					
TEMP									0
TRD-FI (3)							3		
TRD-FP (1)						1			
TRD-FY (1)					1				
AI (*)	1	2	3	1					0
DI									0
MAI								2	0
MDI								3	0
PUL									0
PID	1	1	1	1	1	1	1	1	0
EPID		0	0	0	0	0	0	0	0
APID	0				0	0	0		0
ARTH	1	1	1	1	1	1	1	0	0
SPLT		0	0	0	0	0	1	0	0
CHAR	1	1	1	1	1	1	1	0	0
INTG	1	0	1	1	0	0	1	0	0
AALM	1	2	1	1	1	1	1	1	0

Block type	LD292/ LD302	TT302	IF302	TP302	FY302	FP302	FI302	FB700	DFI302
ISEL	0	1	0	0	1	1	0	0	0
SPG		1	1	1	0	0	0	0	0
TIME	1	0	0	0	0	0	0	0	0
LLAG	1	0	0	0	0	0	0	0	0
OSDL	0	0	0	0	0	0	0	0	0
DENS	0								0
CT	0	0	0	0	0	0	0	0	0
FFET									0
MBCS(16)									0
MBSS(16)									0
MBCM(16)									0
MBSM(16)									0
AO (*)					1	1	3		0
DO									0
MAO								2	0
MDO								4	0
STEP									0

Note 1 – The column “Block type” indicates which block type is available for each type of device.

Note 2 – The number associated to the block type and the device type is the number of blocks instantiated during the factory initialization.

Note 3 – If the function block type is not available to the device type, it will be indicated by blank space.

Note 4 – Field devices and FB700 have a capability of 20 blocks, including resource, transducers and function blocks.

Note 5 – DFI302 has a capability of 100 blocks, including resource, transducers and function blocks.

Note 6 – The column Block type shows the mnemonics, if it is followed by a number between parenthesis, it indicates the maximum number of block instances. If it is followed by “*”, it indicates that maximum number depends on the device type.

2.3 - Resource

RS – Resource Block

Description

This block contains data that is specific to the hardware that is associated with the resource. All data is modeled as Contained, so there are no links to this block. The data is not processed in the way that a function block processes data, so there is no function schematic.

This parameter set is intended to be the minimum required for the Function Block Application associated with the resource in which it resides. Some parameters that could be in the set, like calibration data and ambient temperature, are more part of their respective transducer blocks.

The mode is used to control major states of the resource. O/S mode stops all function block execution. The actual mode of the function blocks will be changed to O/S, but the target mode will not be changed. Auto mode allows normal operation of the resource. IMan shows that the resource is initializing or receiving a software download.

Parameters MANUFAC_ID, DEV_TYPE, DEV_REV, DD_REV, and DD_RESOURCE are required to identify and locate the DD so that Device Description Services can select the correct DD for use with the resource.

The parameter HARD_TYPES is a read only bitstring that indicates the types of hardware that are available to this resource. If an I/O block is configured that requires a type of hardware that is not available, the result will be a block alarm for a configuration error.

The RS_STATE parameter contains the operational state of the Function Block Application for the resource containing this resource block.

RESTART parameter

The RESTART parameter allows degrees of initialization of the resource. They are:

- 1 - Run: it is the passive state of the parameter
- 2 - Restart resource: it is intended to clear up problems like garbage collection
- 3 - Restart with defaults: it is intended to wipe configuration memory, it works like a factory initialization.
- 4 - Restart processor: it provides a way to hit the reset button on the processor associated with the resource

This parameter does not appear in a view because it returns to 1 shortly after being written.

Non-volatile parameters

The Smar devices do not support cyclic saving of non-volatile parameters to a non-volatile memory, therefore NV_CYCLE_T parameter will always be zero, which means not supported feature.

On the other hand, the Smar devices have a mechanism to save non-volatile parameters into a non-volatile memory during the power down, and they will be recovered in the power up.

Timeout for remote cascade modes

SHED_RCAS and SHED_ROUT set the time limit for loss of communication from a remote device. These constants are used by all function blocks that support a remote

cascade mode . The effect of a timeout is described in Mode Calculation. Shedding from RCAS/ROUT shall not happen when SHED_RCAS or SHED_ROUT is set to zero.

Alert Notification

The MAX_NOTIFY parameter value is the maximum number of alert reports that this resource can have sent without getting a confirmation, corresponding to the amount of buffer space available for alert messages. A user can set the number lower than that, to control alert flooding, by adjusting the LIM_NOTIFY parameter value. If LIM_NOTIFY is set to zero, then no alerts are reported. The CONFIRM_TIME parameter is the time for the resource to wait for confirmation of receipt of a report before trying again. If the CONFIRM_TIME = 0 the device shall not retry.

FEATURES / FEATURE_SEL parameters

The bit strings FEATURES and FEATURE_SEL determine optional behavior of the resource. The first defines the available features, and is read only. The second is used to turn on an available feature by configuration. If a bit is set in FEATURE_SEL that is not set in FEATURES, the result will be a block alarm for a configuration error.

Smar devices support the following features: Reports supported, Fault State supported, Soft Write lock supported.

Fault state for the whole resource

If the user sets the SET_FSTATE parameter, the FAULT_STATE parameter will indicate active and it will cause **all output function blocks** in the resource to go immediately to the condition chosen by the fault state Type I/O option. It may be cleared by setting the CLR_FSTATE parameter. The set and clear parameters do not appear in a view because they are momentary.

Write lock by software

The WRITE_LOCK parameter, if set, will prevent any external change to the static or nonvolatile data base in the Function Block Application of the resource. Block connections and calculation results will proceed normally, but the configuration will be locked. It is set and cleared by writing to the WRITE_LOCK parameter. Clearing WRITE_LOCK will generate the discrete alert WRITE_ALM, at the WRITE_PRI priority. Setting WRITE_LOCK will clear the alert, if it exists.

Features being implemented

The parameter CYCLE_TYPE is a bitstring that defines the types of cycles that this resource can do. CYCLE_SEL allows the configurator to choose one of them. If CYCLE_SEL contains more than one bit, or the bit set is not set in CYCLE_TYPE, the result will be a block alarm for a configuration error. MIN_CYCLE_T is the manufacturer specified minimum time to execute a cycle. It puts a lower limit on the scheduling of the resource.

MEMORY_SIZE declares the size of the resource for configuration of function blocks, in kilobytes.

The parameter FREE_SPACE shows the percentage of configuration memory that is still available. FREE_TIME shows the approximate percentage of time that the resource has left for processing new function blocks, should they be configured.

BLOCK_ERR

The BLOCK_ERR of the resource block will reflect the following causes:
Device Fault State Set – When FAULT_STATE is active.
Simulate Active – When the Simulate jumper is ON.
Out of Service – When the block is in O/S mode.

Supported Modes

O/S and AUTO

Parameters

Index	Parameter	DataType (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
1	ST_REV	Unsigned16		0	None	S	
2	TAG_DESC	OctString(32)		Spaces	Na	S	
3	STRATEGY	Unsigned16		0	None	S	
4	ALERT_KEY	Unsigned8	1 to 255	0	None	S	
5	MODE_BLK	DS-69		O/S	Na	S	See Mode Parameter
6	BLOCK_ERR	Bitstring(2)			E	D / RO	
7	RS_STATE	Unsigned8			E	D / RO	State of the function block application state machine.
8	TEST_RW	DS-85			None	D	Read/write test parameter - used only for conformance testing.
9	DD_RESOURCE	VisibletString (32)		Spaces	Na	S / RO	String identifying the tag of the resource which contains the Device Description for this resource.
10	MANUFAC_ID	Unsigned32	Enumeration; controlled by FF	0x00000302	None	S / RO	Manufacturer identification number - used by an interface device to locate the DD file for the resource.
11	DEV_TYPE	Unsigned16	Set by mfrgr		None	S / RO	Manufacturer's model number associated with the resource - used by interface devices to locate the DD file for the resource.
12	DEV_REV	Unsigned8	Set by mfrgr		None	S / RO	Manufacturer revision number associated with the resource - used by an interface device to locate the DD file for the resource.
13	DD_REV	Unsigned8	Set by mfrgr		None	S / RO	Revision of the DD associated with the resource - used by an interface device to locate the DD file for the resource.

Index	Parameter	Data Type (length)	Valid Range/Options	Default Value	Units	Store / Mode	Description
14	GRANT_DENY	DS-70	See Block Options	0	Na	D	Options for controlling access of host computer and local control panels to operating, tuning and alarm parameters of the block.
15	HARD_TYPES	Bitstring(2)	Set by mfr		Na	S / RO	The types of hardware available as channel numbers.
16	RESTART	Unsigned8	1: Run, 2: Restart resource, 3: Restart with defaults, 4: Restart processor		E	D	Allows a manual restart to be initiated. Several degrees of restart are possible.
17	FEATURES	Bitstring(2)	Set by mfr		Na	S / RO	Used to show supported resource block options.
19	FEATURE_SEL	Bitstring(2)		0	Na	S	Used to select resource block options.
19	CYCLE_TYPE	Bitstring(2)	Set by mfr		Na	S / RO	Identifies the block execution methods available for this resource.
20	CYCLE_SEL	Bitstring(2)		0	Na	S	Used to select the block execution method for this resource.
21	MIN_CYCLE_T	Unsigned32	Set by mfr		1/32 millisecc	S / RO	Time duration of the shortest cycle interval of which the resource is capable.
22	MEMORY_SIZE	Unsigned16	Set by mfr		kbytes	S / RO	Available configuration memory in the empty resource. To be checked before attempting a download.
23	NV_CYCLE_T	Unsigned32			1/32 millisecc	S / RO	Interval between writing copies of NV parameters to non-volatile memory. Zero means never.
24	FREE_SPACE	Float	0 to 100 %		%	D / RO	Percent of memory available for further configuration. Zero in a preconfigured resource.
25	FREE_TIME	Float	0 to 100%		%	D / RO	Percent of the block processing time that is free to process additional blocks.
26	SHED_RCAS	Unsigned32		640000	1/32millisec	S	Time duration at which to give up on computer writes to function block RCas locations.
27	SHED_ROUT	Unsigned32		640000	1/32 millisecc	S	Time duration at which to give up on computer writes to function block ROut locations.

Index	Parameter	Data Type (length)	Valid Range / Options	Default Value	Units	Store / Mode	Description
28	FAULT_STATE	Unsigned8	1: Clear, 2: Active		E	D	Condition set by loss of communication to an output block, failure promoted to an output block or a physical contact. When Fault State condition is set, Then output function blocks will perform their FSAFE actions.
29	SET_FSTATE	Unsigned8	1: Off, 2: Set	1	E	D	Allows the fault state condition to be manually initiated by selecting Set.
30	CLR_FSTATE	Unsigned8	1: Off, 2: Clear	1	E	D	Writing a Clear to this parameter will clear the device fault state if the field condition, if any, has cleared.
31	MAX_NOTIFY	Unsigned8	Set by mfr		None	S / RO	Maximum number of unconfirmed notify messages possible.
32	LIM_NOTIFY	Unsigned8	0 to MAX_NOTIFY	MAX_NOTIFY	None	S	Maximum number of unconfirmed alert notify messages allowed.
33	CONFIRM_TIME	Unsigned32		640000	1/32 millisecc	S	The minimum time between retries of alert reports.
34	WRITE_LOCK	Unsigned8	1: Unlocked, 2: Locked	1	E	S	If set, no writes from anywhere are allowed, except to clear WRITE_LOCK. Block inputs will continue to be updated.
35	UPDATE_EVT	DS-73			Na	D	This alert is generated by any change to the static data.
36	BLOCK_ALM	DS-72			Na	D	The block alarm is used for all configuration, hardware, connection failure or system problems in the block. The cause of the alert is entered in the subcode field. The first alert to become active will set the Active status in the Status attribute. As soon as the Unreported status is cleared by the alert reporting task, another block alert may be reported without clearing the Active status, if the subcode has changed.
37	ALARM_SUM	DS-74			Na	S	The current alert status, unacknowledged states, unreported states, and disabled states of the alarms associated with the function block.
38	ACK_OPTION	Bitstring(2)	0: Auto ACK Disable 1: Auto ACK Enable	0	Na	S	Selection of whether alarms associated with the block will be automatically acknowledged.

Index	Parameter	DataType (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
39	WRITE_PRI	Unsigned8	0 to 15	0	None	S	Priority of the alarm generated by clearing the write lock.
40	WRITE_ALM	DS-72			None	D	This alert is generated if the write lock parameter is cleared.

Legend: E – Enumerated parameter; na – Adimensional parameter; RO – Read only; D – dynamic; N – non-volatile; S - static

2.4 - TRANSDUCER BLOCKS

DIAG – Diagnostics Transducer Block

Description

This transducer block provides the following features:
Online measurement of block execution time
Hardware revision
Firmware revision
Serial number of device
Serial number of main board

The parameter INIT_PRM will define which initial values for parameters will be used after a block instantiation. The option READY_TO_USE selects a more suitable initial value set, it will avoid invalid values for parameters. It is still possible to have the initial values defined by specification by selecting the option SPEC.

Supported modes

O/S and AUTO.

Diagnostics Transducer Block Access

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
1	ST_REV	Unsigned16		0	None	S	
2	TAG_DESC	OctString(32)		Spaces	Na	S	
3	STRATEGY	Unsigned16		0	None	S	
4	ALERT_KEY	Unsigned8	1 to 255	0	None	S	
5	MODE_BLK	DS-69		O/S	Na	S	See Mode Parameter
6	BLOCK_ERR	Bitstring(2)			E	D	
7	EXE_TIME_TAG	Visible String (32)		spaces	Na	D	Block tag of the selected block to measure the execution time.
8	MIN_EXE_TIME	Float		+INF	ms	D / RO	Minimum execution time of the selected block.
9	CUR_EXE_TIME	Float		0	ms	D / RO	Current execution time of the selected block.
10	MAX_EXE_TIME	Float		0	ms	D / RO	Maximum execution time of the selected block.
11	HW_REV	Visible String(5)				S / RO	Hardware revision.
12	FIRMWARE_REV	Visible String(5)				S / RO	Firmware revision.
13	DEV_SN	Unsigned32				S / RO	Device serial number.
14	MAIN_BOARD_SN	Unsigned32				S / RO	Main board serial number.

Index	Parameter	Data Type (length)	Valid Range/Options	Default Value	Units	Store / Mode	Description
15	BEHAVIOR	Unsigned8	0:Adapted 1:Spec	0	E	S	Select the initial values for parameters, there are two options READY_TO_USE and SPEC.
16	PUB_SUB_STATU S	Unsigned8	0-good 1-bad		E	D / RO	Indicate if all external links is good or if at least one is bad.
17	LINK_SELECTION	Unsigned8	0-first 1-next 2-previous	0	E	D	Select an external link.
18	LINK_NUMBER	Unsigned16				D / RO	Number of the external link selected.
19	LINK_STATUS	Unsigned8				D / RO	Status of the external link selected
20	LINK_RECOVER	Unsigned8	0-no action 1-action	No action	E	D	Command a recovery process to the external link selected.
21	BLOCK_ALM	DS-72			Na	D	The block alarm is used for all configuration, hardware, connection failure or system problems in the block. The cause of the alert is entered in the subcode field. The first alert to become active will set the Active status in the Status attribute. As soon as the Unreported status is cleared by the alert reporting task, another block alert may be reported without clearing the Active status, if the subcode has changed.
22	SAVING_CONFIG	UNSIGNED 8	0 – NOT SAVING 1 - SAVING	0	E	D	It indicates if the device is saving the configuration in a non-volatile memory.

Legend: E – Enumerated parameter; na – Adimensional parameter; RO – Read only; D – dynamic; N – non-volatile; S - static

Link Status	General Status	Publisher/Subs criber	Connection Status	Sending/Receiving	Block Update
0x00	Good	Publisher			
0x40	Good	Subscriber			
0x84	Bad	Publisher	Established	Sending/Receiving	Not updating
0x88	Bad	Publisher	Established	Not sending/receiving	Updating
0x8c	Bad	Publisher	Established	Not sending/receiving	Not updating
0x98	Bad	Publisher	Not established	Not sending/receiving	Updating
0x9c	Bad	Publisher	Not established	Not sending/receiving	Not updating
0xa8	Bad	Publisher	Pending	Not sending/receiving	Updating
0xac	Bad	Publisher	Pending	Not sending/receiving	Not updating
0xbc	Bad	Publisher	Not configured	Not sending/receiving	Not updating
0xc4	Bad	Subscriber	Established	Sending/Receiving	Not updating
0xcc	Bad	Subscriber	Established	Not sending/receiving	Not updating
0xdc	Bad	Subscriber	Not established	Not sending/receiving	Not updating
0xec	Bad	Subscriber	Pending	Not sending/receiving	Not updating
0xfc	Bad	Subscriber	Not configured	Not sending/receiving	Not updating

0 –Good
Bad

Publisher
Subscriber

0-Established
1-Not established
2-Pending
3-Not configured

0-Sending/Receiving
1-Not sending/receiving

0-Updating
1-Not updating

Display Transducer

Description

The display transducer is responsible to show on the LCD screen, one choosed variable when it is in monitoring mode or a configured menu when in local adjustment mode. The display transducer is completely configured via SYSCON. It means, the user can select the best options to fit his application. Among the possibilities, the following options can be emphasized: Mode block, Outputs monitoring, Tag visualization and Tuning Parameters setting. There are seven groups of parameters, wich should be configured by the user in order to enable the display transducer, which are explained below.

Supported Modes

OOS and AUTO.

Parameters

Index	Parameter Mnemonic	DataType (length)	Valid Range/ Options	Default Value	Units	Store	Description
7	BLOCK_TAG_PARAM	VisibleString			None	S	This is a tag of the block to which the parameter belongs to use up to a maximum of 32 character.
8	INDEX_RELATIVE	Unsigned16	0-65535		None	S	This is the index related to the parameter to be actuated or viewed (1, 2, ..).
9	SUB_INDEX	Unsigned8	1-255		None	S	To visualize a certain tag, opt for the index relative equal to zero, and for the sub-index equal to one.
10	MNEMONIC	VisibleString			None	S	This is the mnemonic for the parameter identification (maximum of 16 characters). Choose the mnemonic, preferably with no more than 5 characters because, this way, it will not necessary to rotate it on display.
11	INC_DEC	Float			None	S	It is the increment and decrement in decimal units when the parameter is Float or Float Status time, or integer, when the parameter is in whole units.
12	DECIMAL_POINT_NUMBER	Unsigned8	0-4		None	S	This is the number of digits after the decimal point (0 to 3 decimal digits)
13	ACCESS	Unsigned8	Monit/Action		None		The access allows the user to read, in the case of the "Monitoring" option, and to write when "action" option is selected, then the display will show the increment and decrement arrows.
14	ALPHA_NUM	Unsigned8	Mnem/Value		None	S	These parameters include two options: value and mnemonic. In option value it is possible to display data both in the alphanumeric and in the numeric fields, this way, in the case of data higher than 10000, it will be shown in the alphanumeric field.

Index	Parameter Mnemonic	DataType (length)	Valid Range/ Options	Default Value	Units	Store	Description
63	DISPLAY_REFLESH	Unsigned8	1		None	D	

HC – Hardware Configuration Transducer

Overview

It configures the module type for each slot in the DFI302.

Description

The following table shows the available module types.

Código	Descrição	I/O Type
	Available slot	No I/O
DF51	DFI302 Processor 1x10Mbps, 4xH1	No I/O
DF50	Power Supply 90-264VAC	No I/O
DF56	Power Supply for Backplane 20-30VDC	No I/O
DF52	Power Supply for Fieldbus	No I/O
DF49	2-channel Power Supply Impedance	No I/O
DF53	4-channel Power Supply Impedance	No I/O
DF11	2 Groups of 8 24VDC Inputs (Isolated)	16-discrete input
DF12	2 Groups of 8 48VDC Inputs (Isolated)	16- discrete input
DF13	2 Groups of 8 60VDC Inputs (Isolated)	16- discrete input
DF14	2 Groups of 8 125VDC Inputs (Isolated)	16- discrete input
DF15	2 Groups of 8 24VDC Inputs (Sink)(Isolated)	16- discrete input
DF16	2 Groups of 4 120VAC Inputs (Isolated)	8- discrete input
DF17	2 Groups of 4 240VAC Inputs (Isolated)	8- discrete input
DF18	2 Groups of 8 120VAC Inputs (Isolated)	16- discrete input
DF19	2 Groups of 8 240VAC Inputs (Isolated)	16- discrete input
DF20	1 Group of 8 On/Off Switches	8- discrete input
DF21	1 Group of 16 Open Collector Outputs	16- discrete output
DF22	2 Group of 8 Transistor Outputs (source) (Isolated)	16- discrete output
DF23	2 Groups of 4 120/240VAC Outputs	8- discrete output
DF24	2 Groups of 8 120/240VAC Outputs	16- discrete output
DF25	2 Groups of 4 NO Relays Outputs	8- discrete output
DF26	2 Groups of 4 NC Relays Outputs	8- discrete output
DF27	1 Group of 4 NO and 4 NC Relay Outputs	8- discrete output
DF28	2 Groups of 8 NO Relays Outputs	16- discrete output
DF29	2 Groups of 4 NO Relays Outputs (W/o RC)	8- discrete output
DF30	2 Groups of 4 NC Relays Outputs (W/o RC)	8- discrete output
DF31	1 Group of 4 NO and 4 NC Relay Outputs (W/o RC)	8- discrete output
DF32	1 Group of 8 24VDC Inputs and 1 Group of 4 NO Relays	8- discrete input/4- discrete output
DF33	1 Group of 8 48VDC Inputs and 1 Group of 4 NO Relays	8- discrete input/4- discrete output

Código	Descrição	I/O Type
DF34	1 Group of 8 60VDC Inputs and 1 Group of 4 NO Relays	8- discrete input/4- discrete output
DF35	1 Group of 8 24VDC Inputs and 1 Group of 4 NC Relays	8- discrete input/4- discrete output
DF36	1 Group of 8 48VDC Inputs and 1 Group of 4 NC Relays	8- discrete input/4- discrete output
DF37	1 Group of 8 60VDC Inputs and 1 Group of 4 NC Relays	8- discrete input/4- discrete output
DF38	1 Group of 8 24VDC Inputs ,1 Group of 2 NO and 2 NC Relays	8- discrete input/4- discrete output
DF39	1 Group of 8 48VDC Inputs , 1 Group of 2 NO and 2 NC Relays	8- discrete input/4- discrete output
DF40	1 Group of 8 60VDC Inputs , 1 Group of 2 NO and 2 NC Relays	8- discrete input/4- discrete output
DF41	2 Groups of 8 pulse inputs – low frequency	16-pulse input
DF42	2 Groups of 8 pulse inputs – high frequency	16-pulse input
DF43	1 Group of 8 analog Inputs	8-analog input
DF44	1 Group of 8 analog inputs with shunt resistors	8-analog input
DF57	1 Group of 8 differential analog inputs with shunt resistors	8-analog input
DF45	1 Group of 8 temperature Inputs	8-temperature
DF46	1 Group of 4 analog output	4-analog output

The execution method of this transducer block will write to all output modules and it will read all the input modules. If any I/O module has failed in this scan, it will be indicated in BLOCK_ERR as well in the MODULE_STATUS_x , It makes easy to find the module in failure or even the sensor.

All the I/O modules in the previous table may be accessed directly using Input/Output Function Blocks without a transducer block, except for the DF-45 that requires the TEMP block.

BLOCK_ERR

The BLOCK_ERR of the HC block will reflect the following causes:

Lost static data – Low battery voltage indication
 Device needs maintenance now – High temperature in the CPU
 Input Failure – a physical input point in failure
 Output Failure – a physical output point in failure
 Out of Service – When the block is in O/S mode.

Supported Modes

O/S and AUTO.

Status Handling

Parameters

Index	Parameter	Data Type (length)	Valid Range/Options	Default Value	Units	Store / Mode	Description
1	ST_REV	Unsigned16		0	None	S	
2	TAG_DESC	OctString(32)		Spaces	Na	S	
3	STRATEGY	Unsigned16		0	None	S	
4	ALERT_KEY	Unsigned8	1 to 255	0	None	S	
5	MODE_BLK	DS-69		O/S	Na	S	See Mode Parameter
6	BLOCK_ERR	Bitstring(2)			E	D / RO	
7	REMOTE_IO	Unsigned8	0 : Master 1 : Remote I/O Slave 1 2 : Remote I/O Slave 2 3 : Remote I/O Slave 3 4 : Remote I/O Slave 4 5 : Remote I/O Slave 5 6 : Remote I/O Slave 6	0	E	S / O/S	Identification for master or slave remote I/O.
8	IO_TYPE_R0	4 Unsigned8		0	E	S / O/S	Select module type for the rack 0
9	IO_TYPE_R1	4 Unsigned8		0	E	S / O/S	Select module type for the rack 1
10	IO_TYPE_R2	4 Unsigned8		0	E	S / O/S	Select module type for the rack 2
11	IO_TYPE_R3	4 Unsigned8		0	E	S / O/S	Select module type for the rack 3
12	IO_TYPE_R4	4 Unsigned8		0	E	S / O/S	Select module type for the rack 4
13	IO_TYPE_R5	4 Unsigned8		0	E	S / O/S	Select module type for the rack 5
14	IO_TYPE_R6	4 Unsigned8		0	E	S / O/S	Select module type for the rack 6
15	IO_TYPE_R7	4 Unsigned8		0	E	S / O/S	Select module type for the rack 7
16	IO_TYPE_R8	4 Unsigned8		0	E	S / O/S	Select module type for the rack 8
17	IO_TYPE_R9	4 Unsigned8		0	E	S / O/S	Select module type for the rack 9
18	IO_TYPE_R10	4 Unsigned8		0	E	S / O/S	Select module type for the rack 10
19	IO_TYPE_R11	4 Unsigned8		0	E	S / O/S	Select module type for the rack 11
20	IO_TYPE_R12	4 Unsigned8		0	E	S / O/S	Select module type for the rack 12
21	IO_TYPE_R13	4 Unsigned8		0	E	S / O/S	Select module type for the rack 13
22	IO_TYPE_R14	4 Unsigned8		0	E	S / O/S	Select module type for the rack 14

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
23	MODULE_STATU S_R0_3	Bitstring(2)				D / RO	Status of modules in rack 0-3.
24	MODULE_STATU S_R4_7	Bitstring(2)				D / RO	Status of modules in rack 4-7.
25	MODULE_STATU S_R8_11	Bitstring(2)				D / RO	Status of modules in rack 8-11.
26	MODULE_STATU S_R12_14	Bitstring(2)				D / RO	Status of modules in rack 12-14.
27	UPDATE_EVT	DS-73			Na	D	This alert is generated by any change to the static data.
28	BLOCK_ALM	DS-72			Na	D	The block alarm is used for all configuration, hardware, connection failure or system problems in the block. The cause of the alert is entered in the subcode field. The first alert to become active will set the Active status in the Status attribute. As soon as the Unreported status is cleared by the alert reporting task, another block alert may be reported without clearing the Active status, if the subcode has changed.

Legend: E – Enumerated parameter; na – Adimensional parameter; RO – Read only; D – dynamic; N – non-volatile;
S - static

If BEHAVIOR parameter is “Adapted”:

MBCF – ModBus Configuration

Parameters

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
1	ST_REV	Unsigned16		0	None	S	
2	TAG_DESC	OctString(32)		Spaces	Na	S	
3	STRATEGY	Unsigned16		0	None	S	
4	ALERT_KEY	Unsigned8	1 to 255	0	None	S	
5	MODE_BLK	DS-69		O/S	Na	S	See Mode Parameter
6	BLOCK_ERR	Bitstring(2)			E	D / RO	
7	MEDIA	Unsigned8	0:Serial, 1:TCP/IP	Serial	E	S / O/S	Define the type of Modbus channel.
8	MASTER_SLAVE	Unsigned8	0:Master, 1:Slave	Slave	E	S / O/S	Define if DFI is master or slave.
9	DEVICE_ADDRESS	Unsigned8	0-247	1	E	S / O/S	Define the DFI Modbus address (only for DFI slave).
10	BAUD_RATE	Unsigned8	0:110, 1:300, 2:600, 3:1200, 4:2400, 5:4800, 6:9600, 7:19200, 8:38400, 9:57600, 10:115200	19200	E	S / O/S	Define the baud rate (only for media serial).
11	STOP_BITS	Unsigned8	0:1, 1:2	1	E	S / O/S	Define the number of stop bits (only for media serial).
12	PARITY	Unsigned8	0:None, 1:Even, 2:Odd.	Even	E	S / O/S	Define the parity (only for media serial).
13	TIMEOUT	Unsigned16	0-65535	1000	ms	S / O/S	Time to wait a response from a slave (for DFI master) or time to wait the OUTs be updated (for DFI slave). Value 0 is used to disable.
14	NUMBER_RETRANSMISSIONS	Unsigned8	0-255	1		S / O/S	Number of retransmission if DFI doesn't receive response from slave.
15	SLAVE_ADDRESSES	DS-263				S / O/S	IP number and modbus addresses of slaves (only for DFI master in TCP/IP media);
16	RESTART_MODBUS	Boolean		FALSE		S / O/S	Indicate if after a communication fail with slave, there will be a new transmission after the time defined in TIME_TO_RESTART (only for DFI master).
17	TIME_TO_RESTART	Unsigned16	1-65535	1	s	S / O/S	Time to restart communication with slave
18	RTS_CTS	Boolean		FALSE		S / O/S	Enable or not handshaking.
19	ON_APPLY	Unsigned16	0:None, 1: Apply	None	E	S / O/S	Apply the changes made in the modbus blocks.
20	UPDATE_EVT	DS-73			Na	D	This alert is generated by any change to the static data.
21	BLOCK_ALM	DS-72			Na	D	The block alarm is used for all configuration, hardware, connection failure or system problems in the block. The cause of the alert is entered in the subcode field. The first alert to become active will set the Active status in the Status attribute. As soon as the Unreported status is cleared by the alert reporting task, another block alert may be reported without clearing the Active status, if the subcode has changed.

2.5 - INPUT TRANSDUCER BLOCKS

LD292 / LD302 - Pressure Transducer

Description

The pressure transducer makes the corrected pressure sensor reading PRIMARY_VALUE available to the AI block. The engineering unit and the primary value range are selected from the XD_SCALE in the AI block. The units allowed are: Pa, KPa, MPa, bar, mbar, torr, atm, psi, g/cm², kg/cm², inH2O a 4°C, inH2O a 68°F, mmH2O a 68°F, mmH2O a 4°C, ftH2O a 68°F, inHg a 0°C, mmHg a 0°C. The XD_SCALE range must be inside the sensor range in the unit selected. Note that the XD_SCALE should be used to cancel out wet-legs etc. instead of calibration. The supported mode is OOS and AUTO. As the transducer block runs together with AI block, the transducer block goes to AUTO only if the AI mode block is already in AUTO. The sensor temperature may be read from the SECONDARY_VALUE parameter.

Warning messages may appear in Primary Value status or in the Block Error in certain condition as explain below.

Supported Modes

OOS and AUTO.

Block_Err

The BLOCK_ERR of the transducer block will reflect the following causes:

Input Failure – When sensor is disconnected from main electronic board, or the pressure is higher or lower 27.5% of the sensor limit.

Out of Service – When the block is in OOS mode.

Primary_Value Status

The PRIMARY_VALUE status of the transducer block will reflect the following causes:

Bad::SensorFailure:NotLimited – When sensor is disconnected from main electronic board, or the pressure is higher or lower 27.5% of the sensor limit.

Uncertain::SensorConversionNotAccurate:NotLimited – when pressure is between 27.5% of sensor limit and the sensor limit.

Parameters

Index	Parameter Mnemonic	Data Type (length)	Valid Range/ Options	Default Value	Units	Store	Description
5	MODE_BLK	DS-69	OOS, AUTO	OOS	None	D	Indicates the operation mode of Transducer Block.
6	BLOCK_ERR	Bitstring(2)			None	S	Indicates the status associated with hardware or software in the Transducer.
14	PRIMARY_VALUE	DS-65			XD_SCALE	D	The measured value and status available to the Function Block.
15	PRIMARY_RANGE_RANGE	DS-68			XD_SCALE	S	The High and Low range limit values, the engineering unit code and the number of digits to the right of the decimal point to be used for Primary Value.
29	SECONDARY_VALUE	DS-65	(-40) –100°C		SVU	D	The secondary value, related to the sensor
30	SECONDARY_VALUE_UNIT	Unsigned16	°C, °F, °R,K	°C	E	S	The engineering units to be used with SECONDARY_VALUE.

TT302 - Temperature Transducer

Description

The temperature transducer makes the direct corrected input or linearized temperature sensor reading PRIMARY_VALUE available to the AI block. The sensor type, the connection and the calculation type are configured at SENSOR_TYPE, SENSOR_CONNECTION and PRIMARY_VALUE_TYPE respectively. Note that when two sensor are being used (i.e. backup, differential or double) the only sensor connection available is two wires. The engineering unit and the primary value range are selected from the XD_SCALE in the AI block. The units allowed are: Ohm for resistance sensor, mV for millivoltage sensor and °C, °F, °R, K for temperature sensor. The XD_SCALE range must be inside the sensor range in the unit selected. The selection of transducer number is done in SENSOR_TRANSDUCER_NUMBER. The second transducer will exist only when the sensor connection is double two wires. In this case two sensors will generate inputs for two transducers. When the sensor works as a backup the second sensor will generate the input if the first one fails. When the sensor works as a differential, the output is the difference of the two inputs. The AI block connected to this transducer has the CHANNEL the same selection as SENSOR_TRANSDUCER_NUMBER. The supported mode is OOS and AUTO. As the transducer block runs together with AI block, the transducer block goes to AUTO only if the AI mode block is already in AUTO. The cold-junction temperature may be read from the SECONDARY_VALUE parameter.

Warning messages may appear in Primary Value status or in the Block Error in certain condition as explain below.

Supported Modes

OOS and AUTO.

Block_Err

The BLOCK_ERR of the transducer block will reflect the following causes:

Input Failure – When the sensor is broken or the sensor reading is out of limits

Out of Service – When the block is in OOS mode.

Primary_Value Status

The PRIMARY_VALUE status of the transducer block will reflect the following causes:

Bad::SensorFailure:NotLimited – When the sensor is broken or the sensor reading is out of limits

Parameters

Index	Parameter Mnemonic	Data Type (length)	Valid Range/Options	Default Value	Units	Store	Description
5	MODE_BLK	DS-69	OOS, AUTO	OOS	None	D	Indicates the operation mode of Transducer Block.
6	BLOCK_ERR	Bitstring(2)			None	S	Indicates the status associated with hardware or software in the Transducer.
13	PRIMARY_VALUE_TYPE	Unsigned16	Backup, Differential temp., Process temp.	Process temp.	None	S	The type of measurement represented by the primary value
14	PRIMARY_VALUE	DS-65			XD_SCALE	D	The measured value and status available to the Function Block.

Index	Parameter Mnemonic	Data Type (length)	Valid Range/Options	Default Value	Units	Store	Description
15	PRIMARY_RANGE_RANGE	DS-68		850 – (-200) °C	XD_SCALE	S	The High and Low range limit values, the engineering unit code and the number of digits to the right of the decimal point to be used for Primary Value.
20	SENSOR_TYPE	Unsigned16	See table below	Pt 100 IEC	PVU	S	The type of sensor.
27	SENSOR_CONNECTION	Unsigned8	See table below	3	None	S	The number of wires for the temperature probe.
28	SECONDARY_VALUE	DS-65	(-40) – 100°C		SVU	D	The secondary value, related to the sensor
29	SECONDARY_UNIT	Unsigned16	°C, °F, °R, K	°C	E	S	The engineering units to be used with SECONDARY_VALUE.
39	SENSOR_TRANSDUCER_NUMBER	Unsigned8	1,2	0	None	S	Select the transducer number

Sensor Type	Class	Sensor range – 2-wire (Celsius)	Sensor range – Differential (Celsius)
Cu 10 GE	RTD	-20 to 250	-270 to 270
Ni 120 DIN		-50 to 270	-320 to 320
Pt 50 IEC		-200 to 850	-1050 to 1050
Pt 100 IEC		-200 to 850	-1050 to 1050
Pt 500 IEC		-200 to 450	-650 to 650
Pt 50 JIS		-200 to 600	-800 to 800
Pt 100 JIS		-200 to 600	-800 to 800
0 to 100	Ohm	0 to 100	-100 to 100
0 to 400		0 to 400	-400 to 400
0 to 2000		0 to 2000	-2000 to 2000
B NBS	TC	100 to 1800	-1700 to 1700
E NBS		-100 to 1000	-1100 to 1100
J NBS		-150 to 750	-900 to 900
K NBS		-200 to 1350	-1550 to 1550
N NBS		-100 to 1300	-1400 to 1400
R NBS		0 to 1750	-1750 to 1750
S NBS		0 to 1750	-1750 to 1750
T NBS		-200 to 400	-600 to 600
L DIN		-200 to 900	-1100 to 1100
U DIN		-200 to 600	-800 to 800
-6 to 22	MV	-6 to 22	-28 to 28
-10 to 100		-10 to 100	-110 to 110
-50 to 500		-50 to 500	-550 to 550

IF302 - Current Fieldbus Transducer

Description

The current fieldbus transducer makes the current input reading PRIMARY_VALUE available to the AI block. The engineering unit and the primary value range are selected from the XD_SCALE in the AI block. The only unit allowed in this case is mA. The XD_SCALE must be inside the current range (0-21 mA). When the XD_SCALE range is set to 4 and 20, this makes the transducer follow the NAMUR standard. For different values no status is issued. The selection of the input terminal for this transducer is done in TERMINAL_NUMBER. The AI block connected to this transducer has the CHANNEL the same selection as TERMINAL_NUMBER.

The supported mode is OOS and AUTO. As the transducer block runs together with AI block, the transducer block goes to AUTO only if the AI mode block is already in AUTO.

Warning messages may appear in Primary Value status or in the Block Error in certain condition as explain below.

Supported Modes

OOS and AUTO.

Block_Err

The BLOCK_ERR of the transducer block will reflect the following causes:

Input Failure – When input current higher than 20.7 and XD_SCALE EU100 = 20.0 or input current lower than 3.7 and XD_SCALE EU0 = 4.0.

Out of Service – When the block is in OOS mode.

Primary_Value Status

The PRIMARY_VALUE status of the transducer block will reflect the following causes:

Bad::SensorFailure:NotLimited – When input current higher than 20.7 and XD_SCALE EU100 = 20.0 or input current lower than 3.7 and XD_SCALE EU0 = 4.0.

Uncertain::EngUnitRangeViolation:LowLimited – When input current between 3.7 and 3.98 and XD_SCALE EU0 = 4.0.

Uncertain::EngUnitRangeViolation:HighLimited – When input current between 20.02 and 20.7 and XD_SCALE EU100 = 20.0.

Parameters

Index	Parameter Mnemonic	Data Type (length)	Valid Range/Options	Default Value	Units	Store	Description
5	MODE_BLK	DS-69	OOS, AUTO	OOS	None	D	Indicates the operation mode of Transducer Block.
6	BLOCK_ERR	Bitstring(2)			None	S	Indicates the status associated with hardware or software in the Transducer.
14	PRIMARY_VALUE	DS-65	0-21		XD_SCALE	D	The measured value and status available to the Function Block.
15	PRIMARY_VALUE_RANGE	DS-68	0-21	4-20 mA	XD_SCALE	S	The High and Low range limit values, the engineering unit code and the number of digits to the right of the decimal point to be used for Primary Value.
25	TERMINAL_NUMBER	Unsigned8	1,2,3	0	None	S	Indicates the input terminal number

TP302 – Position Fieldbus Transducer

Description

The position fieldbus transducer makes the position input reading PRIMARY_VALUE available to the AI block. The engineering unit and the primary value range are selected from the XD_SCALE in the AI block. The only unit allowed in this case is %. The AI block connected to this transducer has the CHANNEL the same selection as TERMINAL_NUMBER. The supported mode is OOS and AUTO. As the transducer block runs together with AI block, the transducer block goes to AUTO only if the AI mode block is already in AUTO. The sensor module temperature may be read from the SECONDARY_VALUE parameter.

Warning messages may appear in Primary Value status or in the Block Error in certain conditions as explained below.

Supported Modes

OOS and AUTO.

Block_Err

The BLOCK_ERR of the transducer block will reflect the following causes:

Input Failure – When mechanic module is disconnected from main electronic board.

Out of Service – When the block is in OOS mode.

Primary_Value Status

The PRIMARY_VALUE status of the transducer block will reflect the following causes:

Bad::SensorFailure:NotLimited – When mechanic module is disconnected from main electronic board.

Parameters

Index	Parameter Mnemonic	Data Type (length)	Valid Range/Options	Default Value	Units	Store	Description
5	MODE_BLK	DS-69	OOS, AUTO	OOS	None	D	Indicates the operation mode of Transducer Block.
6	BLOCK_ERR	Bitstring(2)			None	S	Indicates the status associated with hardware or software in the Transducer.
14	PRIMARY_VALUE	DS-65	0-100		XD_SCALE	D	The measured value and status available to the Function Block.
15	PRIMARY_VALUE_RANGE	DS-68	0-100 %	0-100 %	XD_SCALE	S	The High and Low range limit values, the engineering unit code and the number of digits to the right of the decimal point to be used for Primary Value.
25	SECONDARY_VALUE	DS-65	(-40) –100°C		SVU	D	The secondary value, related to the sensor
26	SECONDARY_VALUE_UNIT	Unsigned16	°C, °F, °R,K	°C	E	S	The engineering units to be used with SECONDARY_VALUE.

TEMP – DF-45 Temperature Transducer

Overview

This is the transducer block for the module DF-45, an eight low signal input module for RTD, TC, mV, Ohm.

Description

This transducer block has parameters to configure the eight inputs of low signal, as well an individual status and value in engineering units for each input. Therefore it is enough to configure only the TEMP block if the purpose is to monitor variables.

If the application is a control loop or calculation, it is also necessary to configure an AI or MAI block to address these variables. One important difference for the TEMP block, when using an AI block to access an input : write to VALUE_RANGE_x parameter is disabled. The user must configure the scale in the XD_SCALE parameter of the AI block, that will be copied to the corresponding VALUE_RANGE_x parameter.

BLOCK_ERR

The BLOCK_ERR will reflect the following causes:

Block Configuration Error - When it is not compatible the CHANNEL parameter and HC configuration (DFI302).

Input Failure – At least one input is in failure (DFI302)

Out of Service – When the block is in O/S mode.

Supported Modes

O/S and AUTO.

Status Handling

Parameters

Index	Parameter	DataType (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
1	ST_REV	Unsigned16		0	None	S	
2	TAG_DESC	OctString(32)		Spaces	Na	S	
3	STRATEGY	Unsigned16		0	None	S	
4	ALERT_KEY	Unsigned8	1 to 255	0	None	S	
5	MODE_BLK	DS-69		O/S	Na	S	See Mode Parameter
6	BLOCK_ERR	Bitstring(2)			E	D / RO	
7	CHANNEL	Unsigned16			None	S / O/S	The rack and slot number of the associated DF-45 module coded as RRSXX.
8	TEMP_0	DS-65				D	Temperature of point 0.
9	TEMP_1	DS-65				D	Temperature of point 1.
10	TEMP_2	DS-65				D	Temperature of point 2.
11	TEMP_3	DS-65				D	Temperature of point 3.
12	TEMP_4	DS-65				D	Temperature of point 4.

Index	Parameter	Data Type (length)	Valid Range/Options	Default Value	Units	Store / Mode	Description
13	TEMP_5	DS-65				D	Temperature of point 5.
14	TEMP_6	DS-65				D	Temperature of point 6.
15	TEMP_7	DS-65				D	Temperature of point 7.
16	VALUE_RANGE_0	DS-68		0-100%	VR0	S / O/S	If it is connected to AI block, it is a copy of XD_SCALE. Otherwise the user may write in this scaling parameter.
17	SENSOR_CONNECTION_0	Unsigned8	1 : differential 2 : 2-wire 3 : 3-wire	3	E	S / O/S	Connection of the sensor 0.
18	SENSOR_TYPE_0	Unsigned8	See table below	Pt 100 IEC	E	S / O/S	Type of sensor 0.
19	VALUE_RANGE_1	DS-68		0-100%	VR1	S / O/S	If it is connected to AI block, it is a copy of XD_SCALE. Otherwise the user may write in this scaling parameter.
20	SENSOR_CONNECTION_1	Unsigned8	1 : differential 2 : 2-wire 3 : 3-wire	3	E	S / O/S	Connection of the sensor 1.
21	SENSOR_TYPE_1	Unsigned8	See table below	Pt 100 IEC	E	S / O/S	Type of sensor 1.
22	VALUE_RANGE_2	DS-68		0-100%	VR2	S / O/S	If it is connected to AI block, it is a copy of XD_SCALE. Otherwise the user may write in this scaling parameter.
23	SENSOR_CONNECTION_2	Unsigned8	1 : differential 2 : 2-wire 3 : 3-wire	3	E	S / O/S	Connection of the sensor 2.
24	SENSOR_TYPE_2	Unsigned8	See table below	Pt 100 IEC	E	S / O/S	Type of sensor 2.
25	VALUE_RANGE_3	DS-68		0-100%	VR3	S / O/S	If it is connected to AI block, it is a copy of XD_SCALE. Otherwise the user may write in this scaling parameter.
26	SENSOR_CONNECTION_3	Unsigned8	1 : differential 2 : 2-wire 3 : 3-wire	3	E	S / O/S	Connection of the sensor 3.
27	SENSOR_TYPE_3	Unsigned8	See table below	Pt 100 IEC	E	S / O/S	Type of sensor 3.
28	VALUE_RANGE_4	DS-68		0-100%	VR4	S / O/S	If it is connected to AI block, it is a copy of XD_SCALE. Otherwise the user may write in this scaling parameter.

Index	Parameter	Data Type (length)	Valid Range/Options	Default Value	Units	Store / Mode	Description
29	SENSOR_CONNECTION_4	Unsigned8	1 : differential 2 : 2-wire 3 : 3-wire	3	E	S / O/S	Connection of the sensor 4.
30	SENSOR_TYPE_4	Unsigned8	See table below	Pt 100 IEC	E	S / O/S	Type of sensor 4.
31	VALUE_RANGE_5	DS-68		0-100%	VR5	S / O/S	If it is connected to AI block, it is a copy of XD_SCALE. Otherwise the user may write in this scaling parameter.
32	SENSOR_CONNECTION_5	Unsigned8	1 : differential 2 : 2-wire 3 : 3-wire	3	E	S / O/S	Connection of the sensor 5.
33	SENSOR_TYPE_5	Unsigned8	See table below	Pt 100 IEC	E	S / O/S	Type of sensor 5.
34	VALUE_RANGE_6	DS-68		0-100%	VR6	S / O/S	If it is connected to AI block, it is a copy of XD_SCALE. Otherwise the user may write in this scaling parameter.
35	SENSOR_CONNECTION_6	Unsigned8	1 : differential 2 : 2-wire 3 : 3-wire	3	E	S / O/S	Connection of the sensor 6.
36	SENSOR_TYPE_6	Unsigned8	See table below	Pt 100 IEC	E	S / O/S	Type of sensor 6.
37	VALUE_RANGE_7	DS-68		0-100%	VR7	S / O/S	If it is connected to AI block, it is a copy of XD_SCALE. Otherwise the user may write in this scaling parameter.
38	SENSOR_CONNECTION_7	Unsigned8	1 : differential 2 : 2-wire 3 : 3-wire	3	E	S / O/S	Connection of the sensor 7.
39	SENSOR_TYPE_7	Unsigned8	See table below	Pt 100 IEC	E	S / O/S	Type of sensor 7.

Index	Parameter	Data Type (length)	Valid Range/Options	Default Value	Units	Store / Mode	Description
40	UPDATE_EVT	DS-73			Na	D	This alert is generated by any change to the static data.
41	BLOCK_ALM	DS-72			Na	D	The block alarm is used for all configuration, hardware, connection failure or system problems in the block. The cause of the alert is entered in the subcode field. The first alert to become active will set the Active status in the Status attribute. As soon as the Unreported status is cleared by the alert reporting task, another block alert may be reported without clearing the Active status, if the subcode has changed.

Legend: E – Enumerated parameter; na – Adimensional parameter; RO – Read only; D – dynamic; N – non-volatile; S – static

Code	Sensor Type	Class	Sensor range – Differential (Celsius)	Sensor range – 2-wire (Celsius)	Sensor range – 3-wire (Celsius)
1	Cu 10 GE	RTD	-270 to 270	-20 to 250	-20 to 250
2	Ni 120 DIN		-320 to 320	-50 to 270	-50 to 270
3	Pt 50 IEC		-1050 to 1050	-200 to 850	-200 to 850
4	Pt 100 IEC		-1050 to 1050	-200 to 850	-200 to 850
5	Pt 500 IEC		-270 to 270	-200 to 450	-200 to 450
6	Pt 50 JIS		-850 to 850	-200 to 600	-200 to 600
7	Pt 100 JIS		-800 to 800	-200 to 600	-200 to 600
51	0 to 100	Ohm		0 to 100	0 to 100
52	0 to 400			0 to 400	0 to 400
53	0 to 2000			0 to 2000	0 to 2000
151	B NBS	TC	-1600 to 1600	100 to 1800	
152	E NBS		-1100 to 1100	-100 to 1000	
153	J NBS		900 to 900	-150 to 750	
154	K NBS		-1550 to 1550	-200 to 1350	
155	N NBS		-1400 to 1400	-100 to 1300	
156	R NBS		-1750 to 1750	0 to 1750	
157	S NBS		-1750 to 1750	0 to 1750	
158	T NBS		-600 to 600	-200 to 400	
159	L DIN		-1100 to 1100	-200 to 900	
160	U DIN		-800 to 800	-200 to 600	
201	-6 to 22	MV		-6 to 22	
202	-10 to 100			-10 to 100	
203	-50 to 500			-50 to 500	

If BEHAVIOR parameter is “Adapted”:

- When the configuration of sensor type means a different class, the connection is automatically changed to default (RTD and Ohm – 3-wire, TC and mV – 2-wire).

2.6 - Output Transducer Blocks

FY302 – Fieldbus Positioner Transducer

Description

The fieldbus positioner transducer receives the demanded valve position FINAL_VALUE from the AO block and uses it as a setpoint for a PID servo-positioning algorithm with adjustable gains SERVO_GAIN and SERVO_RESET. The transducer block also makes the corrected actual position sensor reading RETURN available to the AO block. The engineering unit and the final value range are selected from the XD_SCALE in the AO block. The units allowed are: for linear valve % and mm, for rotary valve %,°,rad.

After setting GAIN and RESET an automatic calibration should be done using SETUP to start the valve operation. The supported mode is OOS and AUTO. As the transducer block runs together with AO block, the transducer block goes to AUTO only if the AO mode block is different from OOS. The sensor module temperature may be read from the SECONDARY_VALUE parameter.

Warning messages may appear in Return status or in the Block Error in certain condition as explain below.

Supported Modes

OOS and AUTO.

Block_Err

The BLOCK_ERR of the transducer block will reflect the following causes:

Block Configuration – When the XD_SCALE has an improper range or unit.

Output Failure – When mechanic module is disconnected from main electronic board or no air supply (if FINAL_VALUE is different from 0 or 100%).

Out of Service – When the block is in OOS mode.

Return Status

The RETURN status of the transducer block will reflect the following causes:

Bad::NonSpecific::NotLimited – When mechanic module is disconnected from main electronic board or no air supply (if FINAL_VALUE is different from 0 or 100%).

Parameters

Index	Parameter Mnemonic	DataType (length)	Valid Range/ Options	Default Value	Units	Store	Description
5	MODE_BLK	DS-69	OOS, AUTO	OOS	None	D	Indicates the operation mode of Transducer Block.
6	BLOCK_ERR	Bitstring(2)			None	S	Indicates the status associated with hardware or software in the Transducer.
13	FINAL_VALUE	DS-65			XD_SCALE	D	The requested current and status written by the AO block.
14	FINAL_VALUE_RANGE	DS-68		0-100 %	XD_SCALE	S	The High and Low range limit values, the engineering unit code and the number of digits to the right of the decimal point to be used for Final Value.
18	SERVO_GAIN	Float	1-45	20	None	S	The servo PID gain valve.
19	SERVO_RESET	Float	0-999	4	Min/rep	S	The servo PID reset valve.

Index	Parameter Mnemonic	DataType (length)	Valid Range/ Options	Default Value	Units	Store	Description
28	VALVE_TYPE	Unsigned8	Linear, Rotary	Linear	None	S	The type of the valve.
37	SECONDARY_VALUE	DS-65	(-40) –100°C		SVU	D	The secondary value, related to the sensor
38	SECONDARY_VALUE_UNIT	Unsigned16	°C, °F, °R,K	°C	E	S	The engineering units to be used with SECONDARY_VALUE.
80	SETUP	Unsigned8	Enable/Disable	Disable	None	N	Enable self calibration
83	RETURN	DS-65	0-100		None	D	The return value to the AO block.

FP302 - Fieldbus Pressure Transducer

Description

The fieldbus pressure transducer block is a basic positioner transducer, which means that it is actually only a direct output, without positioning algorithm. The transducer block receives the demanded pneumatic signal output FINAL_VALUE from the AO block and makes the corrected actual position sensor reading RETURN available to the AO block. The engineering unit and the final value range are selected from the XD_SCALE in the AO block. The units allowed are: Pa, KPa, MPa, bar, mbar, torr, atm, psi, g/cm², kg/cm², inH2O a 4°C, inH2O a 68°F, mmH2O a 68°F, mmH2O a 4°C, ftH2O a 68°F, inHg a 0°C, mmHg a 0°C. The XD_SCALE range must be inside the range in the unit selected (3-30 psi). The supported mode is OOS and AUTO. As the transducer block runs together with AO block, the transducer block goes to AUTO only if the AO mode block is different from OOS. The sensor module temperature may be read from the SECONDARY_VALUE parameter.

Warning messages may appear in Return status or in the Block Error in certain condition as explain below.

Supported Modes

OOS and AUTO.

Block_Err

The BLOCK_ERR of the transducer block will reflect the following causes:

Block Configuration – When the XD_SCALE has an improper range or unit.

Output Failure – When mechanic module is disconnected from main electronic board or no air supply.

Out of Service – When the block is in OOS mode.

Return Status

The RETURN status of the transducer block will reflect the following causes:

Bad::NonSpecific:NotLimited – When mechanic module is disconnected from main electronic board or no air supply.

Parameters

Index	Parameter Mnemonic	DataType (length)	Valid Range/ Options	Default Value	Units	Store	Description
5	MODE_BLK	DS-69	OOS, AUTO	OOS	None	D	Indicates the operation mode of Transducer Block.
6	BLOCK_ERR	Bitstring(2)			None	S	Indicates the status associated with hardware or software in the Transducer.
13	FINAL_VALUE	DS-65			XD_SCALE	D	The requested current and status written by the AO block.
14	FINAL_VALUE_RANGE	DS-68	3-30 psi	3-15 psi	XD_SCALE	S	The High and Low range limit values, the engineering unit code and the number of digits to the right of the decimal point to be used for Final Value.

Index	Parameter Mnemonic	DataType (length)	Valid Range/ Options	Default Value	Units	Store	Description
33	SECONDARY_VALUE	DS-65	(-40) – 100°C		SVU	D	The secondary value, related to the sensor
34	SECONDARY_VALUE_UN IT	Unsigned16	°C, °F, °R,K	°C	E	S	The engineering units to be used with SECONDARY_VALUE.

FI302 – Fieldbus Current Transducer

Description

The fieldbus current transducer block is a basic converter transducer, which means that it is actually only a direct output, without positioning algorithm. The transducer block receives the demanded current signal output FINAL_VALUE from the AO block and it makes the actual position/status reading RETURN available to AO block. The engineering unit and the final value range are selected from the XD_SCALE in the AO block. The only unit allowed in this case is mA. The XD_SCALE range must be inside the current range (4-20). The selection of the output terminal for this transducer is done in TERMINAL_NUMBER (1-3). The AO block connected to this transducer has the CHANNEL the same selection as TERMINAL_NUMBER. The supported mode is OOS and AUTO. As the transducer block runs together with AO block, the transducer block goes to AUTO only if the AO mode block is different from OOS.

Warning messages may appear in Return status or in the Block Error in certain condition as explain below.

Supported Modes

OOS and AUTO.

Block_Err

The BLOCK_ERR of the transducer block will reflect the following causes:
Block Configuration – When the XD_SCALE has an improper range or unit.
Output Failure – When the current loop is broken.
Out of Service – When the block is in OOS mode.

Return Status

The RETURN status of the transducer block will reflect the following causes:
Bad::NonSpecific::NotLimited – When the current loop is broken.

Parameters

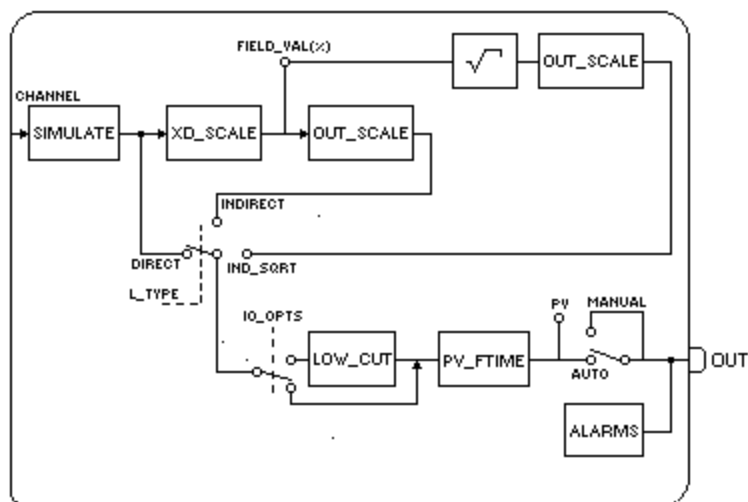
Index	Parameter Mnemonic	DataType (length)	Valid Range/ Options	Default Value	Units	Store	Description
5	MODE_BLK	DS-69	OOS, AUTO	OOS	None	D	Indicates the operation mode of Transducer Block.
6	BLOCK_ERR	Bitstring(2)			None	S	Indicates the status associated with hardware or software in the Transducer.
13	FINAL_VALUE	DS-65	4-20		XD_SCALE	D	The requested current and status written by the AO block.
14	FINAL_VALUE_RANGE	DS-68	4-20 mA	4-20 mA	XD_SCALE	S	The High and Low range limit values, the engineering unit code and the number of digits to the right of the decimal point to be used for Final Value.
24	RETURN	DS-65	4-20		None	D	The return value to the AO block.
25	TERMINAL_NUMBER	Unsigned8	1,2,3	0	None	S	Indicates the input terminal number

2.7 - INPUT FUNCTION BLOCKS

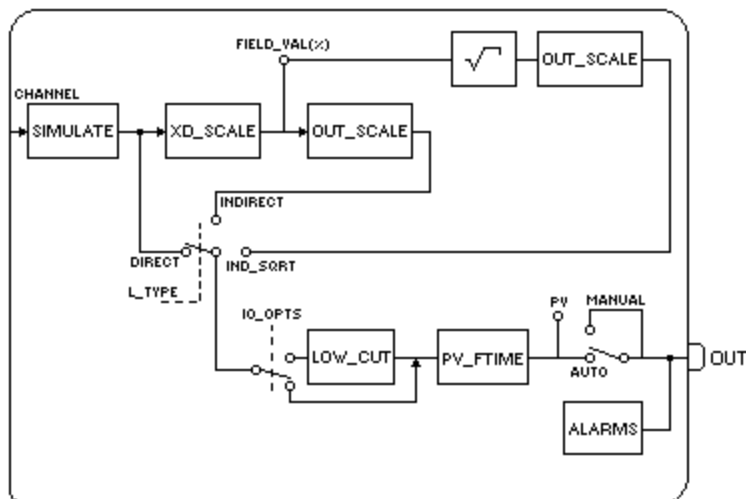
AI - Analog Input

Overview

The Analog Input block takes the input data from the Transducer block, selected by channel number, and makes it available to other function blocks at its output.



Schematic



Description

The AI block is connected to the transducer block through the CHANNEL parameter that must match with the following parameter in the transducer block:

- SENSOR_TRANSDUCER_NUMBER parameter for the TT302
- TERMINAL_NUMBER parameter for the IF302

The CHANNEL parameter must be set to 1 (one) if the AI block is running in the LD302, and no configuration is necessary in the transducer block to connect it to the AI block.

Transducer scaling (XD_SCALE) is applied to the value from the channel to produce the FIELD_VAL in percent. The XD_SCALE engineering units code and range must be suitable to the sensor of transducer block connected to the AI block, otherwise a block alarm indicating configuration error will be generated.

The L_TYPE parameter determines how the values passed by the transducer block will be used into the block. The options are:

Direct - the transducer value is passed directly to the PV. Therefore OUT_SCALE is useless.

Indirect - the PV value is the FIELD_VAL value converted to the OUT_SCALE.

Indirect with Square Root - the PV value is square root of the FIELD_VAL converted to the OUT_SCALE.

PV and OUT always have identical scaling based on OUT_SCALE.

The LOW_CUT parameter is an optional characteristic that may be used to eliminate noise near zero for a flow sensor. The LOW_CUT parameter has a corresponding "Low cutoff" option in the IO_OPTS bit string. If the option bit is true, any calculated output below the low cutoff value (LOW_CUT) will be changed to zero.

BLOCK_ERR

The BLOCK_ERR of the AI block will reflect the following causes:

Block Configuration Error – the configuration error occurs when one or more of the following situations occur:

When the CHANNEL or L_TYPE parameters have an invalid value;

When the XD_SCALE does not have a suitable engineering unit or range for the sensor of transducer block.

When it is not compatible the CHANNEL parameter and HC configuration (DFI302).

Simulate Active – When the Simulate is active.

Input Failure – I/O module failure (DFI302)

Out of Service – When the block is in O/S mode.

Supported Modes

O/S, MAN and AUTO.

Status Handling

The AI block does not support cascade path. Then, The output status has not a cascade sub-status.

When the OUT value exceeds the OUT_SCALE range and no worse condition exists in the block then the OUT status will be "uncertain, EU Range Violation".

The following options from STATUS_OPTS apply, where Limited refers to the sensor limits: (see in the Function block options to more details about each option)

Propagate Fault Forward

Uncertain if Limited

BAD if Limited

Uncertain if Man mode

Parameters

Index	Parameter	DataType (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
1	ST_REV	Unsigned16		0	None	S	
2	TAG_DESC	OctString(32)		Spaces	Na	S	
3	STRATEGY	Unsigned16		0	None	S	
4	ALERT_KEY	Unsigned8	1 to 255	0	None	S	
5	MODE_BLK	DS-69		O/S	Na	S	See Mode Parameter
6	BLOCK_ERR	Bitstring(2)			E	D / RO	
7	PV	DS-65			PV	D / RO	Process analog value for use in executing the function.
8	OUT	DS-65	OUT_SCALE +/- 10%		OUT	D / Man	The analog value calculated as a result of executing the function.
9	SIMULATE	DS-82	1: Disable ; 2: Active are the Enable/Disable options.	Disable		D	Allows the input value to be manually supplied when simulate is enabled. In this case, the simulate value and status will be the PV value.
10	XD_SCALE	DS-68		0-100%	XD	S / Man	The high and low scale values, to transducer for a specified channel.
11	OUT_SCALE	DS-68		0-100%	OUT	S / Man	The high and low scale values to the OUT parameter.

Index	Parameter	Data Type (length)	Valid Range/Options	Default Value	Units	Store / Mode	Description
12	GRANT_DENY	DS-70		0	na	D	
13	IO_OPTS	Bitstring(2)	See Block Options	0	na	S / O/S	See Block Options
14	STATUS_OPTS	Bitstring(2)	See Block Options	0	Na	S / O/S	See Block Options
15	CHANNEL	Unsigned16		0	None	S / O/S	The number of the logical hardware channel to the transducer that is connected to this I/O block.
16	L_TYPE	Unsigned8	1: Direct 2: Indirect 3: Indirect Square Root	0	E	S / Man	Determines how the values passed by the transducer block may be used: Directly (Direct) ; with a percent (Indirect) ; or with a percent and with square root (Ind Sqr Root).
17	LOW_CUT	Float	Non-Negative	0	OUT	S	A value of zero percent of scale is used in block processing if the transducer value falls below this limit, in % of scale. This feature may be used to eliminate noise near zero for a flow sensor.
19	PV_FTIME	Float	Non-Negative	0	Sec	S	Time constant of a single exponential filter for the PV, in seconds.
19	FIELD_VAL	DS-65			%	D / RO	Raw value of the field device in percent of the PV range, with a status reflecting the Transducer condition, before signal characterization (L_TYPE) or filtering (PV_FTIME).
20	UPDATE_EVT	DS-73			Na	D	This alert is generated by any change to the static data.
21	BLOCK_ALM	DS-72			Na	D	The block alarm is used for all configuration, hardware, connection failure or system problems in the block. The cause of the alert is entered in the subcode field. The first alert to become active will set the Active status in the Status attribute. As soon as the Unreported status is cleared by the alert reporting task, another block alert may be reported without clearing the Active status, if the subcode has changed.
22	ALARM_SUM	DS-74	See Block Options		Na	S	The current alert status, unacknowledged states unreported states, and disabled states of the alarms associated with the function block.

Index	Parameter	Data Type (length)	Valid Range/Options	Default Value	Units	Store / Mode	Description
23	ACK_OPTION	Bitstring(2)	0: Auto ACK Disable 1: Auto ACK Enable	0	Na	S	Selection of whether alarms associated with the block will be automatically acknowledged
24	ALARM_HYS	Float	0 to 50 %	0.5%	%	S	Alarm hysteresis parameter. In order to clear the alarm the amount the PV must return within the alarm limit plus hysteresis.
25	HI_HI_PRI	Unsigned8	0 to 15			S	Priority of the high high alarm.
26	HI_HI_LIM	Float	OUT_SCALE, +INF	+INF	OUT	S	The setting for high high alarm in engineering units.
27	HI_PRI	Unsigned8	0 to 15			S	Priority of the high alarm.
28	HI_LIM	Float	OUT_SCALE, +INF	+INF	OUT	S	The setting for high alarm in engineering units.
29	LO_PRI	Unsigned8	0 to 15			S	Priority of the low alarm.
30	LO_LIM	Float	OUT_SCALE, -INF	-INF	OUT	S	The setting for low alarm in engineering units.
31	LO_LO_PRI	Unsigned8	0 to 15			S	Priority of the low low alarm.
32	LO_LO_LIM	Float	OUT_SCALE, -INF	-INF	OUT	S	The setting for low low alarm in engineering units.
33	HI_HI_ALM	DS-71			OUT	D	The status for high high alarm and its associated time stamp.
34	HI_ALM	DS-71			OUT	D	The status for high alarm and its associated time stamp.
35	LO_ALM	DS-71			OUT	D	The status for low alarm and its associated time stamp.
36	LO_LO_ALM	DS-71			OUT	D	The status for low low alarm and its associated time stamp.

Legend: E – Enumerated parameter; na – Adimensional parameter; RO – Read only; D – dynamic; N – non-volatile;
S – static

If BEHAVIOR parameter is “Adapted”:

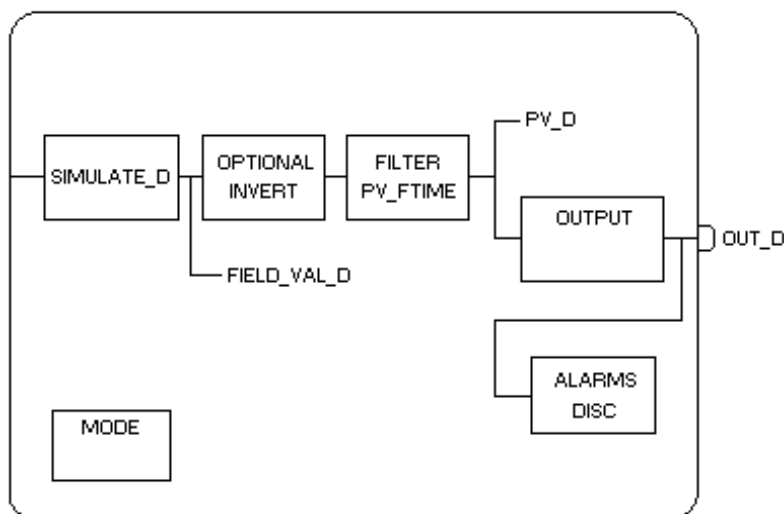
The default value of CHANNEL is the lowest available number.
The default value of L_TYPE is direct.
The required mode for writing is the actual mode, regardless the target mode : OUT

DI - Discrete Input

Overview

The DI block takes the manufacturer's discrete input data, selected by channel number, and makes it available to other function blocks at its output.

Schematic



Description

The FIELD_VAL_D shows the true on/off state of the hardware, using XD_STATE. The Invert I/O option can be used to do a Boolean NOT function between the field value and the output. A discrete value of zero(0) will be considered to be a logical zero(0) and a non-zero discrete value will be considered to be a logical (1) e.g. if invert is selected, the logical NOT of a non-zero field value would result in a zero(0) discrete output, the logical NOT of a zero field value would result in a discrete output value of one(1). PV_FTIME may be used to set the time that the hardware must be in one state before it gets passed to the PV_D. The PV_D is always the value that the block will place in OUT_D if the mode is Auto. If Man is allowed, someone may write a value to OUT_D. The PV_D and the OUT_D always have identical scaling. OUT_STATE provides scaling for PV_D.

BLOCK_ERR

The BLOCK_ERR of the DI block will reflect the following causes:
 Block Configuration Error – the configuration error occurs when one or more of the following situations occur:
 When the CHANNEL parameter has an invalid value;
 When it is not compatible the CHANNEL parameter and HC configuration (DFI302).

Simulate Active – When the Simulate is active.
 Input Failure – I/O module failure (DFI302)

Out of Service – When the block is in O/S mode.

Supported Modes

O/S, Man, and Auto.

Alarm Types

Standard block alarm plus standard discrete alarm applied to OUT_D.

Mode Handling

Standard transition in and out of O/S. See 2.6.2.
Standard transition from Man to Auto and back. Also see 2.6.2.

Status Handling

The status values described in Output Parameter Formal Model of Part 1 apply, with the exception of the control sub-status values. The following options from STATUS_OPTS apply:
Propagate Fault Forward

Initialization

The filter is initialized to the present state of the input.
Function Blocks (Part 2) Fieldbus Specifications FF-891-1.3

Discrete Input Block Access

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
1	ST_REV	Unsigned16		0	None	S	
2	TAG_DESC	OctString(32)		Spaces	Na	S	
3	STRATEGY	Unsigned16		0	None	S	
4	ALERT_KEY	Unsigned8	1 to 255	0	None	S	
5	MODE_BLK	DS-69		O/S	Na	S	See Mode Parameter
6	BLOCK_ERR	Bitstring(2)			E	D / RO	
7	PV_D	DS-66			PV	D / RO	The primary discrete value for use in executing the function, or a process value associated with it.
8	OUT_D	DS-66	OUT_STATE		OUT	D / Man	The primary discrete value calculated as a result of executing the function.
9	SIMULATE_D	DS-83	1: Disable ; 2: Active are the Enable/Disable options.	Disable		D	Allows the transducer discrete input or output to the block to be manually supplied when simulate is enabled. When simulation is disabled, the simulate value and status track the actual value and status.
10	XD_STATE	Unsigned16		0	XD	S	Index to the text describing the states of a discrete for the value obtained from the transducer.
11	OUT_STATE	Unsigned16		0	OUT	S	Index to the text describing the states of a discrete output.

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
12	GRANT_DENY	DS-70		0	na	D	
13	IO_OPTS	Bitstring(2)	See Block Options	0	na	S / O/S	See Block Options
14	STATUS_OPTS	Bitstring(2)	See Block Options	0	Na	S / O/S	See Block Options
15	CHANNEL	Unsigned16		0	None	S / O/S	The number of the logical hardware channel to the transducer that is connected to this I/O block.
16	PV_FTIME	Float	Non-Negative	0	Sec	S	Time constant of a single exponential filter for the PV, in seconds.
17	FIELD_VAL_D	DS-66			On/Off	D / RO	Raw value of the field device discrete input, with a status reflecting the Transducer condition.
18	UPDATE_EVT	DS-73			Na	D	This alert is generated by any change to the static data.
19	BLOCK_ALM	DS-72			Na	D	The block alarm is used for all configuration, hardware, connection failure or system problems in the block. The cause of the alert is entered in the subcode field. The first alert to become active will set the Active status in the Status attribute. As soon as the Unreported status is cleared by the alert reporting task, another block alert may be reported without clearing the Active status, if the subcode has changed.
20	ALARM_SUM	DS-74	See Block Options		Na	S	The current alert status, unacknowledged states, unreported states, and disabled states of the alarms associated with the function block.
21	ACK_OPTION	Bitstring(2)	0: Auto ACK Disable 1: Auto ACK Enable	0	Na	S	Selection of whether alarms associated with the block will be automatically acknowledged
22	DISC_PRI	Unsigned8	0 to 15	0		S	Priority of the discrete alarm.
23	DISC_LIM	Unsigned8	PV_STATE	0	PV	S	State of discrete input which will generate an alarm.
24	DISC_ALM	DS-72			PV	D	The status and time stamp associated with the discrete alarm.

Legend: E – Enumerated parameter; na – Adimensional parameter; RO – Read only; D – dynamic; N – non-volatile;

MAI - Multiple Analog Input

Description

The MAI block makes available for the FF network eight analog variables of the I/O subsystem through its eight output parameters OUT_1 through OUT_8. Status indication in the output parameters OUT_x depends on the I/O subsystem and the transducer block, that is manufacturer specific. For example, if there is individual detection of sensor failure, it will be indicated in the status of related OUT_x parameter. Problem in the interface to the I/O subsystem will be indicated in the status of all OUT_x as BAD – Device Failure.

BLOCK_ERR

The BLOCK_ERR of the MAI block will reflect the following causes:

Other – the number of MDI, MDO, MAI and MAO blocks or the device tag in FB700 is different from LC700.

Block Configuration Error – the configuration error occurs when the OCCURRENCE/CHANNEL has an invalid value (FB700) or it is not compatible the CHANNEL parameter and HC configuration (DFI302).

Input failure – the CPU of LC700 stopped working (FB700) or I/O module failure (DFI302)

Power up – there is no CPU of LC700 in the rack or the hardware configuration of LC700 has an error

Out of Service – When the block is in O/S mode.

Status Handling

The status of OUT_x will be the following if the BLOCK_ERR indicates:

Other – Bad : Configuration Error

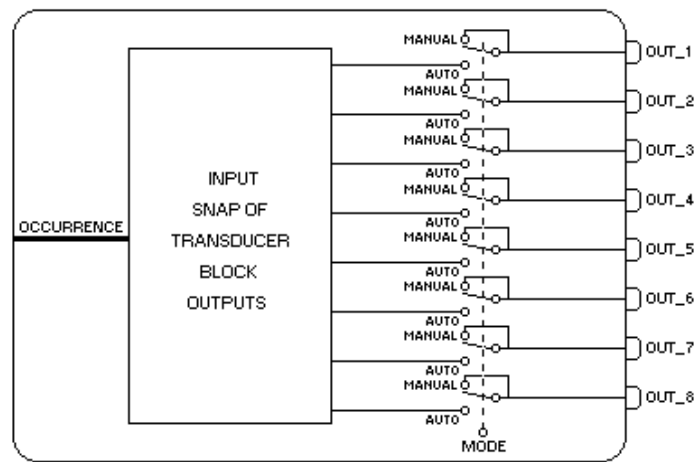
Input failure – Bad : Device Failure

Power up – Bad : Device Failure

Supported Modes

O/S, MAN and AUTO.

Schematic



Parameters

Index	Parameter	DataType (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
1	ST_VER	Unsigned16		0	None	S	
2	TAG_DESC	OctString(32)		Spaces	Na	S	
3	STRATEGY	Unsigned16		0	None	S	
4	ALERT_KEY	Unsigned8	1 to 255	0	None	S	
5	MODE_BLK	DS-69		O/S	Na	S	See Mode Parameter
6	BLOCK_ERR	Bitstring(2)			E	D / RO	
7	OCCURRENCE / CHANNEL	Unsigned16		0	None	S / O/S	The number of the logical hardware OCCURRENCE/CHANNEL that is connected to this multiple I/O block. It defines the transducer to be used going to or from the physical world. It addresses a group of eight points.
8	OUT_1	DS-65				D / Man	Numbered analog input 1.
9	OUT_2	DS-65				D / Man	Numbered analog input 2.
10	OUT_3	DS-65				D / Man	Numbered analog input 3.
11	OUT_4	DS-65				D / Man	Numbered analog input 4.
12	OUT_5	DS-65				D / Man	Numbered analog input 5.
13	OUT_6	DS-65				D / Man	Numbered analog input 6.
14	OUT_7	DS-65				D / Man	Numbered analog input 7.
15	OUT_8	DS-65				D / Man	Numbered analog input 8.
16	BLOCK_ALM	DS-72			Na	D	
17	UPDATE_EVT	DS-73			Na	D	

Legend: E – Enumerated parameter; na – Adimensional parameter; RO – Read only; D – dynamic;
N – non-volatile; S - static

If BEHAVIOR parameter is “Adapted”:

The default value of OCCURRENCE is the number of MAI blocks instantiated in the profile revision 0

The required mode for writing is the actual mode, regardless the target mode : OUT_1, OUT_2, ..., OUT_8.

Device type	Profile revision	Description
FB700	0	Block has OCCURRENCE parameter
DFI302	1	Block has CHANNEL parameter

MDI - Multiple Discrete Input

Description

The MDI block makes available for the FF network eight discrete variables of the I/O subsystem through its eight output parameters OUT_D1 through OUT_D8. Status indication in the output parameters OUT_Dx depends on the I/O subsystem and the transducer block, that is manufacturer specific. For example, if there is individual detection of sensor failure, it will be indicated in the status of related OUT_Dx parameter. Problem in the interface to the I/O subsystem will be indicated in the status of all OUT_Dx as BAD – Device Failure.

BLOCK_ERR

The BLOCK_ERR of the MDI block will reflect the following causes:

Other – the number of MDI, MDO, MAI and MAO blocks or the device tag in FB700 is different from LC700.

Block Configuration Error – the configuration error occurs when the OCCURRENCE has an invalid value (FB700) or it is not compatible the CHANNEL parameter and HC configuration (DFI302).

Input failure – the CPU of LC700 stopped working (FB700) or I/O module failure (DFI302)

Power up – there is no CPU of LC700 in the rack or the hardware configuration of LC700 has an error

Out of Service – When the block is in O/S mode.

Status Handling

The status of OUT_Dx will be the following if the BLOCK_ERR indicates:

Other – Bad : Configuration Error

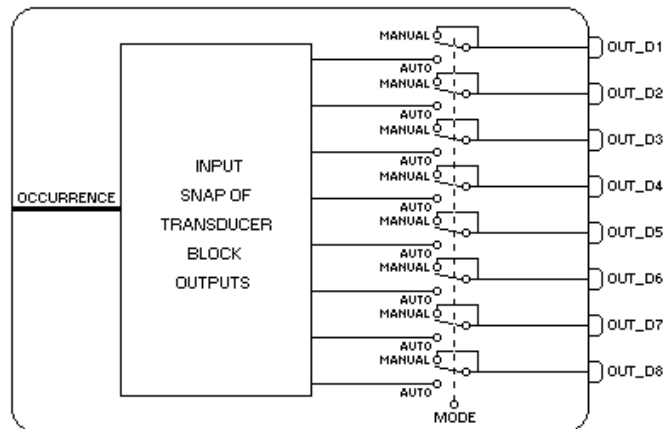
Input failure – Bad : Device Failure

Power up – Bad : Device Failure

Supported Modes

O/S, MAN and AUTO.

Schematic



Parameters

Index	Parameter	DataType (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
1	ST_REV	Unsigned16		0	None	S	
2	TAG_DESC	OctString(32)		Spaces	Na	S	
3	STRATEGY	Unsigned16		0	None	S	
4	ALERT_KEY	Unsigned8	1 to 255	0	None	S	
5	MODE_BLK	DS-69		O/S	Na	S	See Mode Parameter
6	BLOCK_ERR	Bitstring(2)			E	D / RO	
7	OCCURRENCE / CHANNEL	Unsigned16		0	None	S / O/S	The number of the logical hardware OCCURRENCE / CHANNEL that is connected to this multiple I/O block. It defines the transducer to be used going to or from the physical world. It addresses a group of eight points.
8	OUT_D1	DS-66				D / Man	Numbered discrete input 1.
9	OUT_D2	DS-66				D / Man	Numbered discrete input 2.
10	OUT_D3	DS-66				D / Man	Numbered discrete input 3.
11	OUT_D4	DS-66				D / Man	Numbered discrete input 4.
12	OUT_D5	DS-66				D / Man	Numbered discrete input 5.
13	OUT_D6	DS-66				D / Man	Numbered discrete input 6.
14	OUT_D7	DS-66				D / Man	Numbered discrete input 7.
15	OUT_D8	DS-66				D / Man	Numbered discrete input 8.
16	BLOCK_ALM	DS-72			Na	D	
17	UPDATE_EVT	DS-73			Na	D	

Legend: E – Enumerated parameter; na – Admensional parameter; RO – Read only; D – dynamic; N – non-volatile; S - static

If **BEHAVIOR** parameter is “Adapted”:

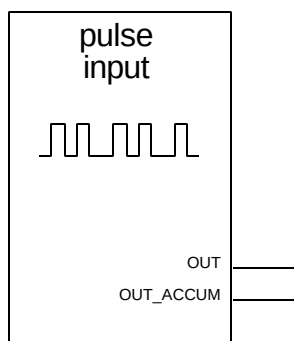
The default value of **OCCURRENCE** is the number of MAI blocks instantiated in the profile revision 0.

Device type	Profile revision	Description
FB700	0	Block has OCCURRENCE parameter
DFI302	1	Block has CHANNEL parameter

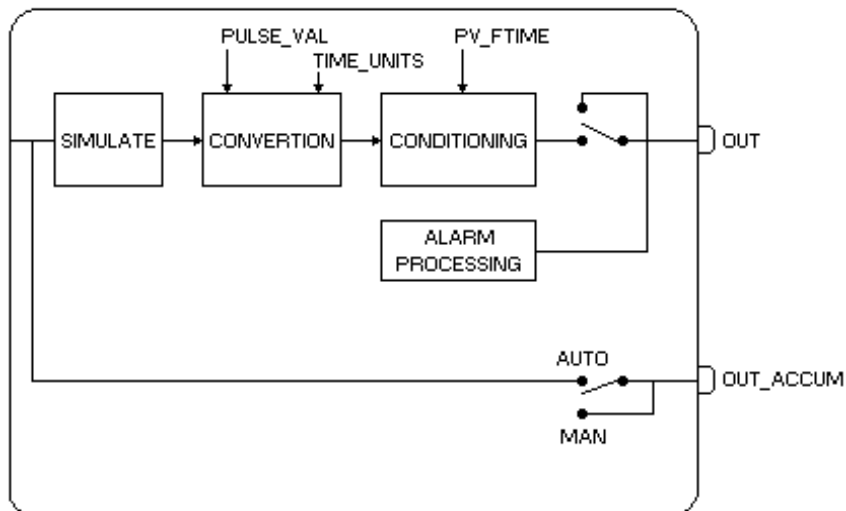
PUL – Pulse Input

Overview

The Pulse Input Block provides analog values based on a pulse (counter) transducer input. There are two primary outputs available. An accumulation output is intended to be connected to an integrator block for differencing, conversion, and integration. This is most useful when the count rate is low relative to the block execution rate. For high count rates, the accumulated count of pulses per block execution can be interpreted as an analog rate (vs. accumulation) value and can be alarmed. (Alarm conditions include high, high-high, low, and low-low alarms.)



Schematic



Description

OUT is a connectable bipolar (signed) analog value output of the PI block. It is determined by taking the number of counts accumulated since the last execution of the block, multiplying by the value of each pulse (PULSE_VAL), dividing by the block's execution rate in seconds, converting to units/minute, units/hour, or units/day in accordance with the TIME_UNITS enumeration, and filtering using PV_FTIME. PV_FTIME is the time constant for a filter. Alarming is performed on this filtered value. Reverse flow may be detected from the transducer and indicated via a negative value of OUT.

Pre-filtered value = (change_in_counts * PULSE_VAL / exec_period) * time_unit_factor
Where,

CHANGE_IN_COUNTS is the number of counts received since last execution

PULSE_VAL is the value in engineering units of each pulse

EXEC_PERIOD is the execution period of the block in seconds time_unit_factor is 1 sec/sec, 60 sec/min, 3600 sec/hour, or 86400 sec/day, per the TIME_UNITS index.

For example, 70 pulse counts are received by the transducer in the 0.5 second execution period of the PI block. The metering device manufacturer specifies that each pulse represents 0.1 gallons of flow. The user wants the flow rate expressed in "gallons per minute". PULSE_VAL should be set to 0.1. TIME_UNITS should be set to "units/minute". The related time_unit_factor will be "60 sec/min". Using the above equation, the pre-filtered rate value will then be computed as:

$$\begin{aligned} & ((70 \text{ pulses} * 0.1 \text{ gallons/pulse}) / 0.5 \text{ sec}) * 60 \text{ sec/min} = \\ & (7.0 \text{ gallons/} 0.5 \text{ sec}) * 60 \text{ sec/min} = \\ & (14.0 \text{ gallons/sec}) * 60 \text{ sec/min} = \\ & 840 \text{ gallons/min} \end{aligned}$$

OUT_ACCUM is a connectable float output of the PI block. It is intended to be connected to the Integrator block for totalization, so it only accumulates enough to avoid rollover between executions of the Integrator block. It represents a continuous accumulation of counts from the transducer, limited to the range of values from 0 to 999,999. It can count either up or down. An increment of the accumulation 999,999 by 1 will result in the accumulation 0 and a decrement of the accumulation 0 by 1 will result in the accumulation 999,999. A maximum change to the accumulation of $\pm 499,999$ counts is permitted to be reflected in OUT_ACCUM in a single execution of the block. If a change of counts greater in magnitude than 499,999 occurs at the transducer:

- The change in OUT_ACCUM is limited to 499,999 of the proper sign,
- The OUT_ACCUM status's quality is set to uncertain,
- The OUT_ACCUM status's sub-status is set to "Engr. Units Range Violation",
- The OUT_ACCUM status's limits = low (if negative) or limits = high (if positive) indicator is set
- A BLOCK_ALM must be issued.

CHANNEL is used to associate the block with the hardware that is connected to this block. It defines the transducer to be used coming from the physical world.

Supported Modes

O/S, Manual, and Auto modes are supported.

Alarm Types

Standard block alarm plus standard HI_HI, HI, LO, and LO_LO alarms applied to OUT.

Mode Handling

Manual mode "disconnects" the input from the output and permits manual substitution of the value. OUT is the alarmed value and the value which normally would be substituted, but OUT_ACCUM may also be substituted.

On transition from Manual to Auto, the PV filter will be initialized to the value of OUT, and the accumulated total will be set to the value of OUT_ACCUM.

Status Handling

This block has no inputs from other blocks and therefore does not react to status of other blocks. Both the OUT and the OUT_ACCUM outputs have status and will reflect the status of the transducer (e.g. hardware failure) and the mode of the block (e.g., out-of-service, manual, etc.) using the conventional rules of status.

An unusable status (bad) for OUT will cause the alarm processing to be suspended. Current alarms will not be cleared and new alarms will not be generated until the status returns to a usable status.

Initialization

Standard.

Power Failure Recovery

Standard.

Simulation

The SIMULATE_P parameter is provided to simulate pulse input as a rate in pulses/second, rather than the actual transducer value. The value entered in the SIMULATE_P record is considered to be the signed change in accumulation per second. The quality of the status entered is passed to the status of OUT and OUT_ACCUM.

Parameters

Index	Parameter	DataType (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
1	ST_REV	Unsigned16		0	None	S	
2	TAG_DESC	OctString(32)		Spaces	Na	S	
3	STRATEGY	Unsigned16		0	None	S	
4	ALERT_KEY	Unsigned8	1 to 255	0	None	S	
5	MODE_BLK	DS-69		O/S	Na	S	See Mode Parameter
6	BLOCK_ERR	Bitstring(2)			E	D / RO	
7	PV	DS-65			PV	D / RO	Process analog value for use in executing the function.
8	OUT	DS-65	OUT_SCALE +/- 10%		OUT	D / Man	The analog value calculated as a result of executing the function.

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
9	OUT_ACCUM	DS-65			None	N / Man	This parameter is the number of counts accumulated in an on-going basis. It is not normally reset except that it wraps around to zero after reaching 999,999 counts. The value is based on the transducer input in <i>Auto</i> mode and is the last transducer value of the value specified by the operator/engineer in <i>Man</i> mode. (Its meaning is most useful when the number of counts received between executions of the block is small.) It is intended to be connected to the counter input of an integrator block. The OUT_ACCUM value may increase or decrease by a maximum of 499,999 counts per execution.
10	SIMULATE_P	DS-82	1: Disable ; 2: Active are the Enable/Disable options.	Disable		D	Allows the transducer input to the Pulse Input block to be manually supplied when simulate is enabled. When simulation is disabled, the simulate value and status track the actual value and status. The value is the rate of change of the transducer count in counts per second, not the accumulation.
11	PULSE_VAL	Float		0	None		Value of each metered pulse in engineering units. It must have same units as the numerator of the units in OUT_SCALE (e.g., "gallons" per pulse if OUT_SCALE units are "gallons/minute"). Used only to calculate PV and OUT. Not used for OUT_ACCUM calculation.
12	TIME_UNITS	Unsigned8	1: seconds 2: minutes 3: hours 4: days 5: [day- [hr:[min[:sec]]]]	0	E	S	Display Time Units for TIMER_SP, OUT_EXP, and OUT_REM: Seconds, Minutes, Hours, Days, [day- [hr:[min[:sec]]]]
13	OUT_SCALE	DS-68		0-100%	OUT	S / Man	The high and low scale values to the OUT parameter.
14	GRANT_DENY	DS-70		0	na	D	
15	IO_OPTS	Bitstring(2)	See Block Options	0	na	S / O/S	See Block Options
16	STATUS_OPTS	Bitstring(2)	See Block Options	0	Na	S / O/S	See Block Options

Index	Parameter	DataType (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
17	CHANNEL	Unsigned16		0	None	S / O/S	The number of the logical hardware channel to the transducer that is connected to this I/O block.
18	PV_FTIME	Float	Non-Negative	0	Sec	S	Time constant of a single exponential filter for the PV, in seconds.
19	FIELD_VAL	DS-65			%	D / RO	Raw value of the field device with a status reflecting the Transducer condition, before filtering (PV_FTIME).
20	UPDATE_EVT	DS-73			Na	D	This alert is generated by any change to the static data.
21	BLOCK_ALM	DS-72			Na	D	The block alarm is used for all configuration, hardware, connection failure or system problems in the block. The cause of the alert is entered in the subcode field. The first alert to become active will set the Active status in the Status attribute. As soon as the Unreported status is cleared by the alert reporting task, another block alert may be reported without clearing the Active status, if the subcode has changed.
22	ALARM_SUM	DS-74	See Block Options		Na	S	The current alert status, unacknowledged states unreported states, and disabled states of the alarms associated with the function block.
23	ACK_OPTION	Bitstring(2)	0: Auto ACK Disable 1: Auto ACK Enable	0	Na	S	Selection of whether alarms associated with the block will be automatically acknowledged
24	ALARM_HYS	Float	0 to 50 %	0.5%	%	S	Alarm hysteresis parameter. In order to clear the alarm the amount the PV must return within the alarm limit plus hysteresis (percentage of OUT_SCALE).
25	HI_HI_PRI	Unsigned8	0 to 15			S	Priority of the high high alarm.
26	HI_HI_LIM	Float	OUT_SCALE, +INF	+INF	OUT	S	The setting for high high alarm in engineering units.
27	HI_PRI	Unsigned8	0 to 15			S	Priority of the high alarm.
28	HI_LIM	Float	OUT_SCALE, +INF	+INF	OUT	S	The setting for high alarm in engineering units.
29	LO_PRI	Unsigned8	0 to 15			S	Priority of the low alarm.
30	LO_LIM	Float	OUT_SCALE, -INF	-INF	OUT	S	The setting for low alarm in engineering units.
31	LO_LO_PRI	Unsigned8	0 to 15			S	Priority of the low low alarm.

Index	Parameter	DataType (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
32	LO_LO_LIM	Float	OUT_SCALE, - INF	-INF	OUT	S	The setting for low low alarm in engineering units.
33	HI_HI_ALM	DS-71			OUT	D	The status for high high alarm and its associated time stamp.
34	HI_ALM	DS-71			OUT	D	The status for high alarm and its associated time stamp.
35	LO_ALM	DS-71			OUT	D	The status for low alarm and its associated time stamp.
36	LO_LO_ALM	DS-71			OUT	D	The status for low low alarm and its associated time stamp.

Legend: E – Enumerated parameter; na – Adimensional parameter; RO – Read only; D – dynamic; N – non-volatile;
S - static

If BEHAVIOR parameter is “Adapted”:

The default value of CHANNEL is the lowest available number.

The required mode for writing is the actual mode, regardless the target mode : OUT

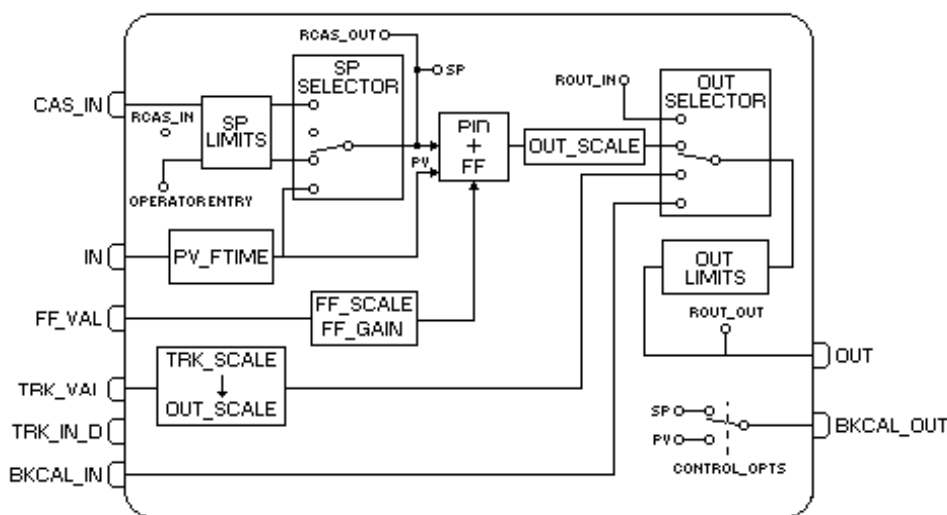
2.8 - CONTROL AND CALCULATION FUNCTION BLOCKS

PID - Pid Control

Overview

The PID block offers a lot of control algorithms that use the Proportional, integral and derivative terms.

Schematic



Description

The algorithm of the PID is the non-iterative or ISA. In this algorithm, the GAIN is applied to all terms of the PID, and the Proportional and the Integral actuate over the error, and the derivative actuates over the PV value. Therefore user changes of SP will not cause bump in the output due to the derivative term when the block is in Auto.

As long as an error exists, the PID function will integrate the error, which moves the output in a direction to correct the error. PID blocks may be cascaded when the difference in process time constants of a primary and secondary process measurement makes it necessary or desirable.

See the PV calculation and SP calculation section for details.

Direct and Reverse Acting

It is possible to choose the direct or reverse action of control that is made through the "Direct Acting" bit in the CONTROL_OPTS parameter:

If the "Direct acting" bit is true then the error is obtained subtracting the SP from the PV:

$$\text{Error} = (\text{PV} - \text{SP})$$

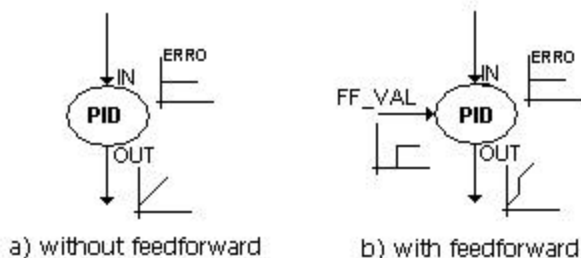
If the "Direct acting" bit is false (clear), the choice is "Reverse acting" then the error is obtained subtracting the PV from the SP:

$$\text{Error} = (\text{SP} - \text{PV})$$

The Default value of the "Direct acting" bit is false, it means "reverse action".

Feedforward Control

The PID block supports the feedforward algorithm. The FF_VAL input is supplied by an external value, which is proportional to some disturbance in the control loop. The value is converted to output scale using the FF_SCALE and OUT_SCALE parameters. This value is multiplied by the FF_GAIN and added to the output of the PID algorithm.



If the status of FF_VAL is Bad, the last usable value will be used. When the status returns to good, the difference of FF_VAL values will be subtracted from BIAS_A/M in order to avoid bump in the output.

PID Constants

GAIN (Kp), RESET (Tr), and RATE (Td) are the tuning constants for the P, I and D terms, respectively. Gain is a dimensionless number. RESET and RATE are time constants expressed in seconds. There are existing controllers that are tuned by the inverse value of some or all of them, such as proportional band and repeats per minute. The human interface to these parameters should be able to display the user's preference.

Bypass

When bypass is active the SP value will be transferred to the OUT without the calculation of PID terms. Bypass is used in secondary cascade controller when the PV is bad.

Conditions to turn the Bypass on:

The "Bypass Enable" bit in the CONTROL_OPTS must be true.

BYPASS parameter is changed to ON.

The BYPASS parameter is the ON/OFF switch that activates the bypass. By default, it can be changed only when the block mode is Man or O/S. Optionally, when the "Change of Bypass in an automatic mode" bit in the FEATURES_SEL parameter in Resource block is true, then the block permits that the BYPASS switch changes in automatic modes too.

There is special treatment when the Bypass parameter changes ON/OFF in order to avoid bump in the output. When the bypass is switched to ON, the SP receives the OUT value in percent of the OUT_SCALE. And when the bypass is switched to OFF, the SP receives the PV value.

Transition in BYPASS	Action
OFF -> ON	OUT -> SP with scaling conversion
ON -> OFF	PV -> SP

Below, there is an example of the bypass in the PID block working as a PID slave in cascade control.

- Step 1 – the status of IN is bad, therefore the actual mode of PID is Man
- Step 2 - the target mode is changed to Man in order to write BYPASS
- Step 3 – the user sets BYPASS to ON, and OUT is transferred to SP with scaling conversion
- Step 4 – the user changes the target mode to Cas
- Step 5 – the PID block reaches the Cas mode, despite of IN.Status.
- Step 7 – the status of IN becomes good
- Step 8 - the target mode is changed to Man in order to write BYPASS
- Step 9 – the user sets BYPASS to OFF, and PV is transferred to SP

CONTROL_OPTS = "Bypass Enable"

Steps	1	2	3	4	5	6	7	8	9	10	11
Target	Cas	Man		Cas				Man		Cas	
Bypass	Off		On						Off		
IN	Bad	Bad	Bad	Bad	Bad	Bad	GNC 80	GNC 80	GNC 80	GNC 80	GNC 80
SP	GC 50	GC 50	GC 20	GC 20	GC 20	GC 20	GC 20	GC 20	GC 80	GC 80	GC 80
Actual	Man	Man	Man	Man	Cas	Cas	Cas	Man	Man	Man	Cas
BKCAL_OUT	NI	NI	NI	IR	GC	GC	GC	NI	NI	IR	GC
OUT	GC 20	GC 20	GC 20	GC 20	GC 20	GC 20	GC 20	GC 20	GC 20	GC 20	GC 20

Legend: GNC-Good Non Cascade status; GC-Good Cascade status

Output Tracking

The PID block supports the output track algorithm, which allows the output to be forced to a tracking value when the tracking switch is on.

In order to activate the output tracking, the block should attend the following conditions:

The "Track Enable" bit in the CONTROL_OPTS must be true.

The target mode is an "automatic" mode (Auto, Cas and Rcas) or Rout.

The TRK_VAL and TRK_IN_D status are usable, it means that the status is good or uncertain with the STATUS_OPTS."Use Uncertain as good" bit true.

The TRK_IN_D value is active.

If the target mode is Man, it is necessary besides the above conditions:

The "Track in Manual" bit in CONTROL_OPTS must be true.

When the output tracking is active, the output OUT will be replaced by the TRK_VAL converted to OUT_SCALE. The output limit status becomes constant and the actual mode goes to LO.

If the TRK_IN_D or TRK_VAL status is unusable, the Output tracking will be off and the PID will return to the normal operation.

Additional features for the Enhanced PID block (EPID)

The EPID function block provides the following additional features:

Different type of transfer from a "manual" mode to an "automatic" mode.

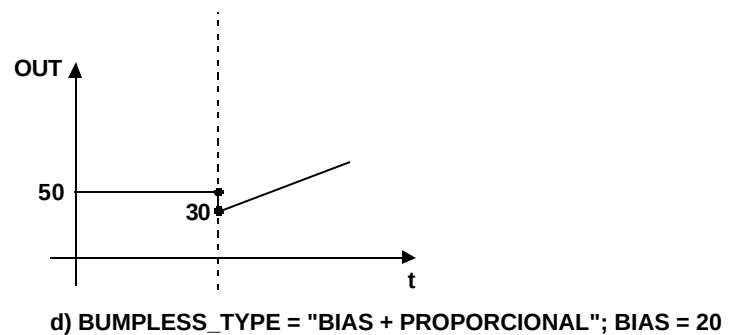
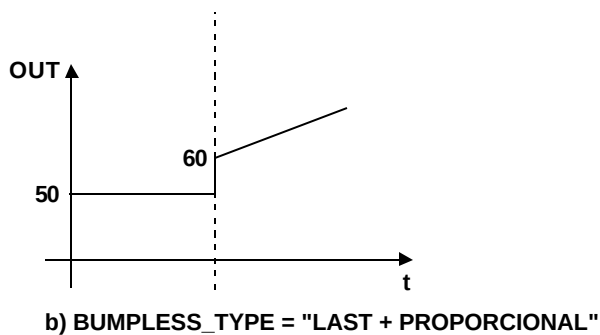
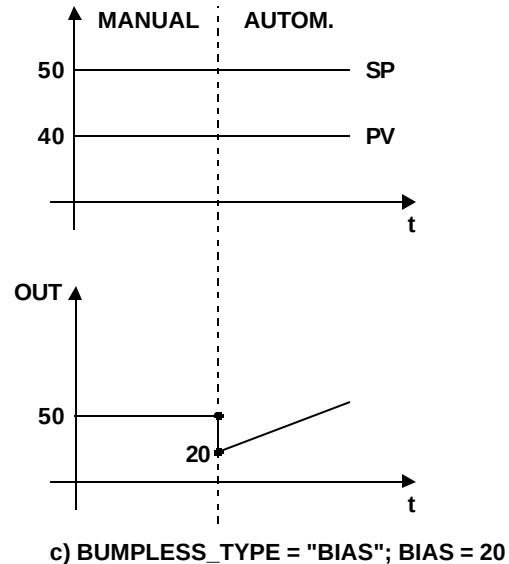
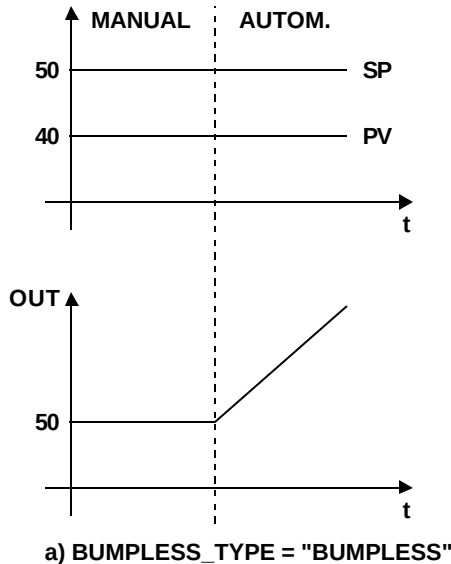
The BUMPLESS_TYPE parameter provides four types of transfer from a "manual" mode to an "automatic" mode:

a. bumpless : This is the default value and the behavior of the standard PID block. The block starts to calculate from the last value in the "manual" mode.

Last + proportional: The block starts to calculate from the last value in the "manual" mode plus the proportional term

b. Bias: The block starts to calculate from the BIAS parameter.

c. Bias + proportional: The block starts to calculate from the BIAS parameter plus the proportional term.



a.Special treatment to Output Tracking

Special treatment is made when the output tracking is enabled:

The algorithm generates an IFS status in the output in the following situations:

When TRK_IN_D has an unusable status and the "IFS if Bad TRK_IN_D" bit in PID_OPTS is true.

When TRK_VAL has an unusable status and the "IFS if Bad TRK_VAL" bit in PID_OPTS is true.

The mode is changed to Man when the tracking inputs are not usable in the following ways:

When the TRK_IN_D is not usable and the 'Man if Bad TRK_IN_D' bit in PID_OPTS is true then the mode will be Man and the OUT will be the last value. Optionally, if the "target to Man if Bad TRK_IN_D" bit in the PID_OPTS is true, then the target mode will be changed to Man too.

When the TRK_VAL is not usable and the "Man if Bad TRK_VAL" bit in PID_OPTS is true, then the mode will be Man and the OUT will be the last usable value. Optionally, if the "target to Man if Bad TRK_VAL" bit in the PID_OPTS is true, then the target mode will be changed to Man too.

The required actions are summarized in the following table:

Situation	PID_OPTS	Mode		Algorithm Action
		Target	Actual	
TRK_IN_D is not usable	0x00		"auto"	. Output tracking is not active. . The algorithm continues the normal calculation.
	IFS if Bad TRK_IN_D		"auto" -> lman	. Output tracking is not active. . The algorithm continues the normal calculation. . OUT.Status is GoodC-IFS. . When the output block goes to fault state, the upper blocks go to lman.
	Man if Bad TRK_IN_D		Man	. Output tracking is not active. . The algorithm stops the calculation.
	"Target to Man if Bad TRK_IN_D" ; "Man if Bad TRK_IN_D"	Man	Man	. Output tracking is not active. The target mode is changed to Man.
TRK_VAL is not usable	0x00		"auto"	. Output tracking is not active. . The algorithm continues the normal calculation.
	IFS if Bad TRK_VAL		"auto" -> lman	. Output tracking is not active. . The algorithm continues the normal calculation. OUT.Status is GoodC-IFS. . When the output block goes to fault state, the upper blocks go to lman.
	Man if Bad TRK_VAL		Man	. Output tracking is not active. . The algorithm stops the calculation.
	"Target to Man if Bad TRK_VAL" ; "Man if Bad TRK_VAL"	Man	Man	. Output tracking is not active. the target mode is changed to Man.
TRK_IN_D and TRK_VAL is usable, TRK_IN_D is active, output tracking is enabled			LO	Output Tracking is active.

If the additional parameters of EPID block is configured with the default values, the block works as the standard PID block.

BLOCK_ERR

The BLOCK_ERR of the PID block will reflect the following causes:

Block Configuration Error – the configuration error occurs when the BYPASS and SHED_OPT parameters have an invalid value;
Out of Service – it occurs when the block is in O/S mode.

Supported Modes

O/S, IMAN, LO, MAN, AUTO, CAS, RCAS and ROUT.

Control Algorithm

$$OUT = GAIN * \left[E + \frac{RATE * S}{1 + \alpha * RATE * S} * PV + \frac{E}{RESET * S} \right] + BIAS\_A/M + FEEDFORWARD$$

NOTE: ① BIAS_A/M: Internal BIAS Calculated on changing to automatic modes (RCAS, CAS, AUTO).

- α : Pseudo - Derivative Gain Equals to 0.13

Parameters

Index	Parameter	Data Type (length)	Valid Range/Options	Default Value	Units	Store / Mode	Description
1	ST_REV	Unsigned16		0	None	S	
2	TAG_DESC	OctString(32)		Spaces	Na	S	
3	STRATEGY	Unsigned16		0	None	S	
4	ALERT_KEY	Unsigned8	1 to 255	0	None	S	
5	MODE_BLK	DS-69		O/S	Na	S	See Mode Parameter
6	BLOCK_ERR	Bitstring(2)			E	D / RO	
7	PV	DS-65			PV	D / RO	Process analog value. This is the IN value after pass over the PV filter.
8	SP	DS-65	PV_SCALE +/- 10%		PV	N / Auto	The analog set point. Can be set manually, automatically through the interface device or another field device.
9	OUT	DS-65	OUT_SCALE +/- 10%		OUT	N / Man	The output value result of the PID calculation.
10	PV_SCALE	DS-68		0-100%	PV	S / Man	The high and low scale values to the PV and SP parameter.
11	OUT_SCALE	DS-68		0-100%	OUT	S / Man	The high and low scale values to the OUT parameter.

Index	Parameter	Data Type (length)	Valid Range/Options	Default Value	Units	Store / Mode	Description
12	GRANT_DENY	DS-70		0	na	D	Options for controlling access of host computer and local control panels to operating, tuning and alarm parameters of the block.
13	CONTROL_OPTS	Bitstring(2)	See Block Options	0	na	S / O/S	See Block Options
14	STATUS_OPTS	Bitstring(2)	See Block Options	0	Na	S / O/S	See Block Options
15	IN	DS-65			PV	D	The primary input value of the block, or PV value.
16	PV_FTIME	Float	Non-Negative	0	Sec	S	Time constant of a single exponential filter for the PV, in seconds.
17	BYPASS	Unsigned8	1:Off 2:On	0	E	S / Man	When bypass is set, the setpoint value (in percent) will be directly transferred to the output.
18	CAS_IN	DS-65				D	This parameter is the remote setpoint value, which must come from another Fieldbus block, or a DCS block through a defined link.
19	SP_RATE_DN	Float	Positive	+INF	PV/Sec	S	Ramp rate at which upward setpoint changes in PV units per second. It is disable if is zero or +INF. Rate limiting will apply only in AUTO mode.
20	SP_RATE_UP	Float	Positive	+INF	PV/Sec	S	Ramp rate at which downward setpoint changes in PV units per second. It is disable if is zero or +INF. Rate limiting will apply only in AUTO mode.
21	SP_HI_LIM	Float	PV_SCALE +/- 10%	100	PV	S	The setpoint high limit is the highest setpoint operator entry that can be used for the block.
22	SP_LO_LIM	Float	PV_SCALE +/- 10%	0	PV	S	The setpoint low limit is the lowest setpoint operator entry that can be used for the block.
23	GAIN	Float		0	None	S	Proportional term of the PID. It is the Kp value.
24	RESET	Float	Positive	+INF	sec	S	Integral term of the PID. It is the Tr value.
25	BAL_TIME	Float	Positive	0	sec	S	This specifies the time for the internal working value of bias or ratio to return to the operator set bias or ratio, in seconds. In the PID block, it may be used to specify the time constant at which the integral term will move to obtain balance when the output is limited and the mode is Auto, Cas, or Rcas.
26	RATE	Float	Positive	0	sec	S	Derivative term of the PID. It is the Td value.

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
27	BKCAL_IN	DS-65			OUT	N	The value and status from a lower block's BKCAL_OUT that is used to prevent reset windup and to initialize the control loop.
28	OUT_HI_LIM	Float	OUT_SCALE +/- 10%	100	OUT	S	Limits the maximum output value.
29	OUT_LO_LIM	Float	OUT_SCALE +/- 10%	0	OUT	S	Limits the minimum output value.
30	BKCAL_HYS	Float	0 to 50%	0.5%	%	S	The amount that the output must change away from its output limit before the limit status is turned off, expressed as a percent of the span of the output.
31	BKCAL_OUT	DS-65			PV	D / RO	The value and status required by an upper block's BKCAL_IN so that the upper block may prevent reset windup and provide bumpless transfer to closed loop control.
32	RCAS_IN	DS-65			PV	D	Target setpoint and status provided by a supervisory Host to a analog control or output block.
33	ROUT_IN	DS-65			OUT	D	Target output and status provided by a Host to the control block for use as the output (Rout mode).
34	SHED_OPT	Unsigned8	1: NormalShed, NormalReturn 2: NormalShed, NoReturn 3: ShedToAuto, NormalReturn 4: ShedToAuto, NoReturn 5: ShedToMan, NormalReturn 6: ShedToMan, NoReturn 7: ShedToRetained Target, NormalReturn 8: ShedToRetained Target, NoReturn	0		S	Defines action to be taken on remote control device timeout.
35	RCAS_OUT	DS-65			PV	D / RO	Block setpoint and status after ramping - provided to a supervisory Host for back calculation and to allow action to be taken under limiting conditions or mode change.

Index	Parameter	Data Type (length)	Valid Range/Options	Default Value	Units	Store / Mode	Description
36	ROUT_OUT	DS-65			OUT	D / RO	Block output and status - provided to a Host for back calculation in ROut mode and to allow action to be taken under limited conditions or mode change
37	TRK_SCALE	DS-68		0-100%	TRK	S / Man	The high and low scale values, engineering units code, and number of digits to the right of the decimal point, associated with TRK_VAL.
38	TRK_IN_D	DS-66			On/Off	D	This discrete input is used to initiate external tracking of the block output to the value specified by TRK_VAL.
39	TRK_VAL	DS-65			TRK	D	This input is used as the track value when external tracking is enabled by TRK_IN_D.
40	FF_VAL	DS-65			FF	D	The feed forward value and status.
41	FF_SCALE	DS-68		0-100%	FF	S	The feedforward input high and low scale values, engineering units code, and number of digits to the right of the decimal point.
42	FF_GAIN	Float		0	none	S/Man	The gain that the feed forward input is multiplied by before it is added to the calculated control output.
43	UPDATE_EVT	DS-73			Na	D	This alert is generated by any change to the static data.
44	BLOCK_ALM	DS-72			Na	D	The block alarm is used for all configuration, hardware, connection failure or system problems in the block. The cause of the alert is entered in the subcode field. The first alert to become active will set the Active status in the Status attribute. As soon as the Unreported status is cleared by the alert reporting task, another block alert may be reported without clearing the Active status, if the subcode has changed.
45	ALARM_SUM	DS-74	See Block Options		Na	S	The current alert status, unacknowledged states, unreported states, and disabled states of the alarms associated with the function block.
46	ACK_OPTION	Bitstring(2)	0: Auto ACK Disable 1: Auto ACK Enable	0	Na	S	Selection of whether alarms associated with the block will be automatically acknowledged

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
47	ALARM_HYS	Float	0 to 50 %	0.5%	%	S	Alarm hysteresis parameter. In order to clear the alarm the amount the PV must return within the alarm limit plus hysteresis.
48	HI_HI_PRI	Unsigned8	0 to 15	0		S	Priority of the high high alarm.
49	HI_HI_LIM	Float	OUT_SCALE, +INF	+INF	PV	S	The setting for high high alarm in engineering units.
50	HI_PRI	Unsigned8	0 to 15	0		S	Priority of the high alarm.
51	HI_LIM	Float	OUT_SCALE, +INF	+INF	PV	S	The setting for high alarm in engineering units.
52	LO_PRI	Unsigned8	0 to 15	0		S	Priority of the low alarm.
53	LO_LIM	Float	OUT_SCALE, -INF	-INF	PV	S	The setting for low alarm in engineering units.
54	LO_LO_PRI	Unsigned8	0 to 15	0		S	Priority of the low low alarm.
55	LO_LO_LIM	Float	OUT_SCALE, -INF	-INF	PV	S	The setting for low low alarm in engineering units.
56	DV_HI_PRI	Unsigned8	0 to 15	0		S	Priority of the deviation high alarm.
57	DV_HI_LIM	Float	0 to PV span, +INF	+INF	PV	S	The setting for deviation high alarm in engineering units.
58	DV_LO_PRI	Unsigned8	0 to 15	0		S	Priority of the deviation low alarm.
59	DV_LO_LIM	Float	-INF, -PV span to 0	-INF	PV	S	The setting for deviation low alarm in engineering units.
60	HI_HI_ALM	DS-71			PV	D	The status for high high alarm and its associated time stamp.
61	HI_ALM	DS-71			PV	D	The status for high alarm and its associated time stamp.
62	LO_ALM	DS-71			PV	D	The status for low alarm and its associated time stamp.
63	LO_LO_ALM	DS-71			PV	D	The status for low low alarm and its associated time stamp.
64	DV_HI_ALM	DS-71			PV	D	The status for deviation high alarm and its associated time stamp.
65	DV_LO_ALM	DS-71			PV	D	The status for deviation low alarm and its associated time stamp.

Enhanced PID - Additional Parameters

Index	Parameter	Type	Valid Range/ Options	Default Value	Units	Mode To Change	Description
66	BUMPLESS_TYPE	Unsigned8	0: Bumpless 1: Last+Proportional 2: Bias 3: Bias+Proportional	0	E	S / Man	Options that defines the algorithm action to start the output when the block transfer from a "manual" to an "automatic" mode.
67	BIAS	Float		0	OUT	S	The bias value to use in the PID algorithm when the BUMPLESS type is "Bias" or "Bias+Proportional".
68	PID_OPTS	Bitstring(2)	See block options	0		S / O/S	The options for handling the additional features of the output tracking.

Legend: E – Enumerated parameter; na – Adimensional parameter; RO – Read only; D – dynamic;
N – non-volatile; S - static

If BEHAVIOR parameter is "Adapted":

The default value of BYPASS is OFF.

The default value of SHED_OPT is NormalShed/NormalReturn.

The required mode for writing is the actual mode, regardless the target mode : SP and OUT.

APID – Advanced Pid

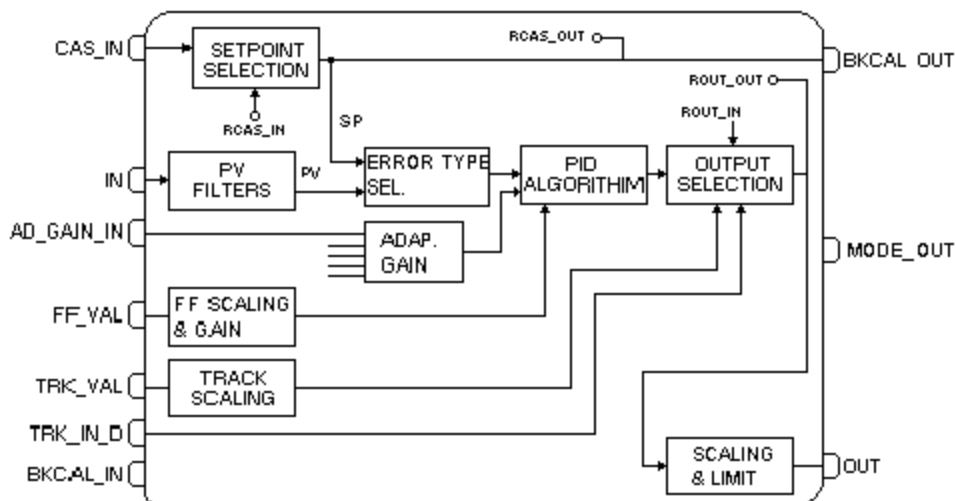
Overview

The advanced PID function block provides the following additional features comparing to the standard PID algorithm and the enhanced PID:

- selection of the terms (proportional, integral, derivative) calculated on error or process variable
- PI Sampling algorithm
- Adaptive gain
- Configurable Limits of anti reset wind-up
- Special treatment for the error
- Discrete output to indicate the actual mode

The standard features as well the enhanced ones are described in the PID block, therefore they will not be repeated here.

Schematic



Description

Selection of the terms (proportional, integral, derivative) calculated on error or process variable

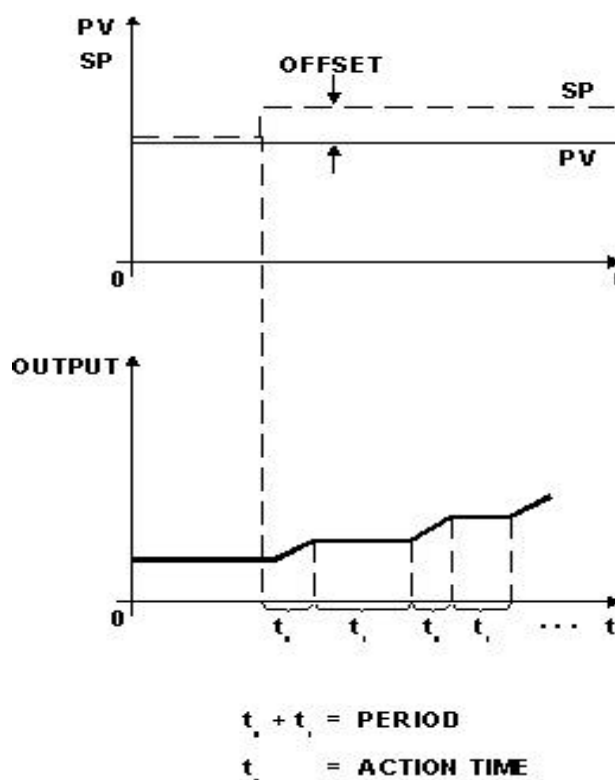
The control algorithm can be ideal parallel or not iterative (ISA). For each algorithm can choose the terms (proportional, integral, and derivative) calculated on error or process variable by setting the PID_TYPE parameter. Where:

PI.D - The P and I terms are calculated based on the error and the D term on the PV.

PID - The P, I and D terms are calculated based on the error.

I.PD - The I is calculated based on the error and the P and D terms on the PV.

PI Sampling algorithm



The output is calculated based on the PI algorithm during time t_i . After that, the algorithm stops calculating and it holds the last value during time t_d . The time t_d is adjusted by `SAMPLE_ON`, and t_i by $(\text{SAMPLE\_PER} - \text{SAMPLE\_ON})$.

If the `SAMPLE_PER` is less than `SAMP_ON` or `SAMP_ON` is zero, then the algorithm works as an ordinary PI controller.

Adaptive gain

The adaptive gain permits to change the algorithm PID terms by a factor obtained in a curve established by `CURVE_X` and `CURVE_Y` parameters. This curve is based in `SP`, `PV`, `Error`, `OUT` or another value set in `AD_GAIN_IN` parameter. The algorithm actions that will be changed are defined by the `AD_GAIN_ACTION` parameter. The `AD_GAIN_IN_SEL` parameter selects the input value to enter into the curve in order to get the adaptive gain.

The `CURVE_X` points of the curve are in the same engineering units of the selected variable. The `CURVE_Y` points are the adaptive gain. The adaptive gain (G) changes the PID constants `GAIN`, `RESET` and `RATE` to:

$$\text{GAIN}' = G * \text{GAIN}$$

$$\text{RESET}' = \text{RESET} / G$$

$$\text{RATE}' = G * \text{RATE}$$

If the curve has less than 20 configured points, the non-configured points shall be set with `+INF`.

If the curve has a configuration error, then the `BLOCK_ALM` indicates it, and the adaptive gain value will be the `CURVE_Y` corresponding to the highest `CURVE_X` point.

If the `AD_GAIN_IN` is selected and it has a bad status, the algorithm uses the last usable

value to provide bumpless transfer.

Configurable Limits of anti reset wind-up

The saturation limits to integral term can be changed by the ARW_LOW and ARW_UP parameters. Then the control algorithm stops the integral calculation when the output signal reaches the anti-reset wind-up limits. The proportional and derivative calculations are not affected.

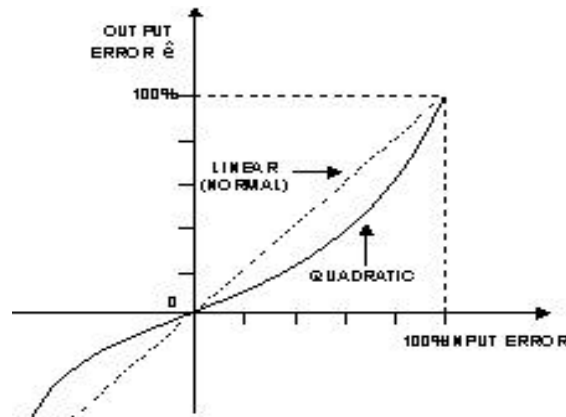
The Anti Reset Wind-up will not be stopped to the output limits, i.e., when the ARW_UP limit is greater than OUT_HI_LIM, the OUT is clamped in the OUT_HI_LIM value, but internally the algorithm continue the integral calculation until the ARW_UP limit. The user can avoid this case configuring the ARW_UP less or equal to OUT_HI_LIM. The same idea applies to the low limit.

Special treatment for the error

The treatment of the error in the control process can be chosen by the ERROR_TYPE parameter.

The quadratic error can be applied on only integral term or on all PID terms. In the quadratic error, the considered error for the calculation will be:

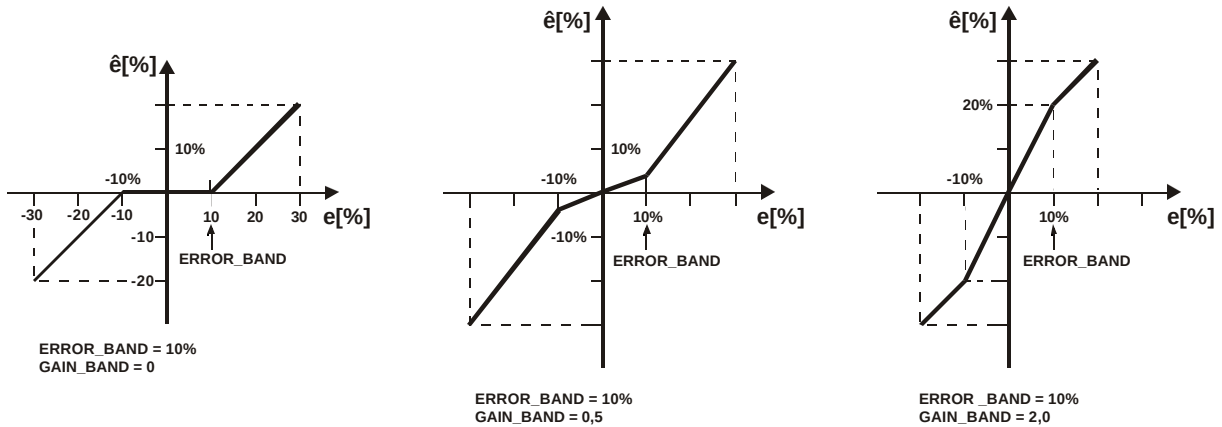
$$\hat{e} = \frac{e^*|e|}{100}$$



In order to use the GAP control to consider the cases where the control is unstable over a band around SP due the dead band of the actuator or due the noise or other things, there is a special gain in the error type.

In order to use the ERROR_TYPE as special gain it is necessary to define the ERROR_BAND where it will be applied the special gain GAIN_BAND parameter on the error. If the ERROR_BAND is zero the algorithm will not apply the special gain.

Mode indication



The MODE_IND parameter is used to configure which mode(s) types in the actual mode will be indicated by a TRUE value in the discrete output MODE_OUT. If more than one mode type is chosen, then it will be used an OR logic.

Working as a standard PID

If the additional parameters of APID block is configured with the default values, then it works as the standard PID block.

Supported modes

O/S, IMAN, LO, MAN, AUTO, CAS, RCAS and ROUT.

Advanced PID Block Access

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
66	MODE_OUT	DS-66				D / RO	This output parameter indicates if one of selected modes in the parameter "MODE_IND" is the actual mode
67	MODE_IND	Bitstring	Actual mode bitstring	0		D	Selection of mode(s) to be indicated in the output parameter "MODE_OUT".
68	AD_GAIN_ACTION	Unsigned8	0: Disable 1: PID 2: PI 3: P 4: I 5: D	0	E	S / Man	It chooses the term(s) of PID algorithm multiplied by the adaptive gain.
69	AD_GAIN_IN_SEL	Unsigned8	0: SP 1: PV 2: Error 3: OUT 4: AD_GAIN_IN	0	E	S / Man	It selects the input to enter into the curve in order to get the adaptive gain. The option Error may be selected if ERROR_TYPE is Normal.

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
70	AD_GAIN_IN	DS-65				D	Input parameter to enter into the curve in order to get the adaptive gain.
71	CURVE_X	20 Floats		0's	EU of variable selected by AD_GAIN_IN_SEL	S	Curve input points. The x _i points of the curve are defined by an array of twenty points.
72	CURVE_Y	20 Floats		0's	Na	S	Curve output points. The y _i points of the curve are defined by an array of twenty points.
73	ERROR_TYPE	Unsigned8	0: Normal 1: Quadratic (Integral) 2: Quadratic (all terms) 3: Special gain	0	E	S / Man	Type of error used by the PID algorithm. The options Quadratic and Special gain may be selected if AD_GAIN_IN_SEL is different of Error.
74	ERROR_BAND	Float	0-300%	0	%	S	It is applied a special treatment for error within the "ERROR_BAND".
75	GAIN_BAND	Float	0-10	0	Na	S	Special gain applied to the error, if it is selected in the ERROR_BAND.
76	PID_TYPE	Unsigned8	0:PI.D + ISA 1:PID + ISA 2:I.PD + ISA 3:PI Sampling + ISA 4:PI.D + Parallel 5:PID + Parallel 6:I.PD + Parallel 7:PI Sampling+Parallel	0	E	S	Type of PID algorithm.
77	SAMP_ON	Float	0-10800	0	Sec	S	Time interval of the PID algorithm activity, therefore (SAMP_PER – SAMP_ON) means the hold time.
78	SAMP_PER	Float	0-10800	0	Sec	S	Period of the PI sampling algorithm.
79	BUMPLESS_TYPE	Unsigned8	0: Bumpless 1: Last+Proportional 2: Bias 3: Bias+Proportional	0	E	S / Man	This parameter defines the type of transfer from a "manual" mode to an "automatic" mode.

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
80	BIAS	Float		0	OUT	S	The bias value to use in the PID algorithm when the BUMPLESS type is "Bias" or "Bias+Proportional".
81	ARW_UP	Float		+ INF	OUT	S	High limit for anti reset windup.
82	ARW_LOW	Float		- INF	OUT	S	Low limit for anti reset windup.
83	PID_OPTS	Bitstring(2)		0		S / O/S	A bit string for handling the additional characteristics of the output tracking.

Legend: *E* – Enumerated parameter; *na* – Adimensional parameter; *RO* – Read only; *D* – dynamic;
N – non-volatile; *S* - static

ARTH - Arithmetic

Description

The ARTH block is intended for use in calculating measurements from combinations of signals from sensors. It is not intended to be used in a control path, so it does not support cascades or back calculation. It does no conversions to percent, so scaling is not supported. It has no process alarms.

The block has 5 inputs. The first two are dedicated to a range extension function that results in a PV, with status reflecting the input in use. The remaining three inputs are combined with the PV in a selection of four term math functions that have been found useful in a variety of measurements. The inputs used to form the PV must come from devices with the desired engineering units, so that the PV enters the equation with the right units. Each of the additional inputs has a bias and gain constant. The bias can be used to correct for absolute temperature or pressure. The gain can be used to normalize terms within a square root function. The output also has gain and bias constants for any further adjustment required.

The range extension function has a graduated transfer, controlled by two constants referenced to IN. An internal value, g, is zero for IN less than RANGE_LO. It is one when IN is greater than RANGE_HI. It is interpolated from zero to one over the range of RANGE_LO to RANGE_HI.

The equation for PV follows:

$$PV = g * IN + (1 - g) * IN\_LO$$

if :

IN < RANGE_LO or

IN_LO < RANGE_HI and status of IN is unusable and status of IN_LO is usable:

g = 0

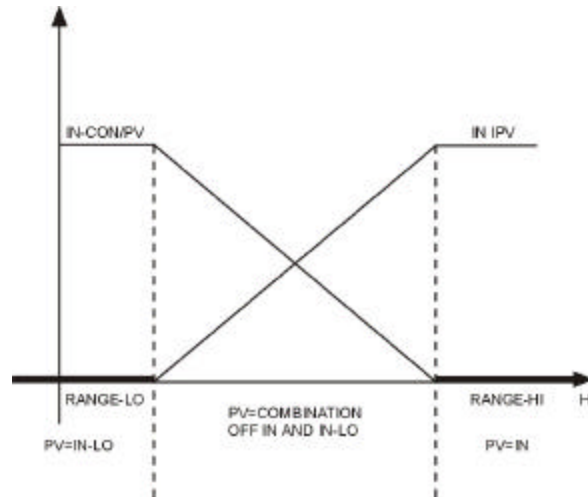
IN > RANGE_HI or

IN > RANGE_LO and status of IN is usable and status of IN_LO is unusable:

g = 1

RANGE_LO ≤ IN ≤ RANGE_HI:

$$g = \frac{IN - RANGE\_LO}{RANGE\_HI - RANGE\_LO}$$



If the status of IN_LO is unusable and IN is usable and greater than RANGE_LO, then g will be set to one. If the status of IN is unusable, and IN_LO is usable and less than RANGE_HI, then g will be set to zero. In each case the PV will have a status of Good until the condition no longer applies. Otherwise, the status of IN_LO is used for the PV if g is less than 0.5, while IN is used for g greater than or equal to 0.5.

Six constants are used for the three auxiliary inputs. Each has a BIAS_IN_i and a GAIN_IN_i. The output has a BIAS and GAIN static constant. For the inputs, the bias is added and the gain is applied to the sum. The result is an internal value called t_i in the function equations.

$$t_i = (IN_i + BIAS\_IN_i) * GAIN\_IN_i$$

The flow compensation functions have limits on the amount of compensation applied to the PV, to assure graceful degradation if an auxiliary input is unstable.

The following equations have the compensation factor limited to COMP_HI_LIM and COMP_LO_LIM:

flow compensation , linear
flow compensation, square root
flow compensation, approximate
BTU flow
Traditional multiply divide

Arithmetic exceptions :

- Division by zero will produce a value equals to OUT_HI_LIM or OUT_LO_LIM, it depends on the sign of PV.
- Roots of negative numbers will produce the root of the absolute value, with a negative sign.

Although the output is not scaled, it still has absolute high and low limits, to keep the values reasonable.

MINIMUM CONFIGURATION

RANGE_HI and RANGE_LO: If the range extension function is not used, these two parameters must be set to -INF. Therefore the PV will be a copy of IN.

If the ARITH_TYPE is one of five first equations, the COMP_HI_LIM and COMP_LO_LIM parameters must be set properly. The default value of COMP_HI_LIM parameter is zero.

As the default value of GAIN parameter is zero, it is necessary to set a suitable value.

BLOCK_ERR

The BLOCK_ERR of the Arithmetic block will reflect the following causes:

Block Configuration Error – the configuration error occurs when the ARITH_TYPE has an invalid value.

Out of Service – When the block is in O/S mode.

Supported Modes

O/S, MAN and AUTO.

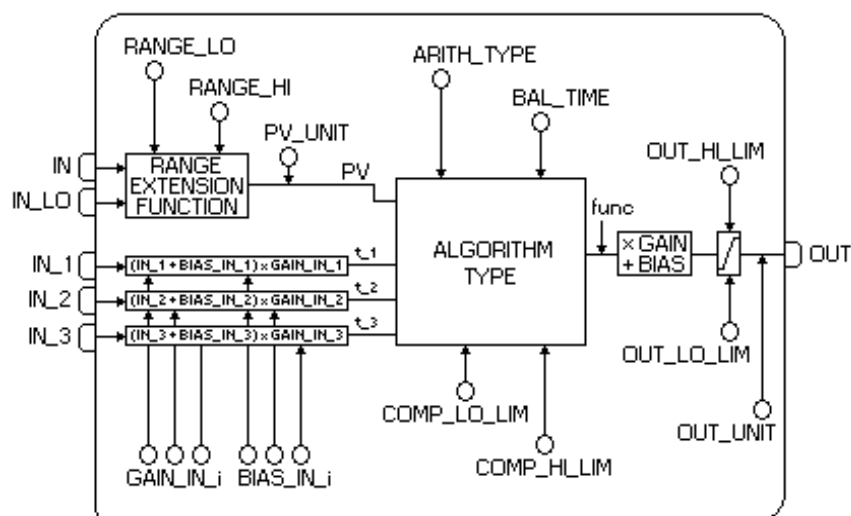
Status Handling

The status of PV depends on the factor “g”, if it is less than 0.5, then it will be used the status of IN_LO, and otherwise it will be the status of IN.

The INPUT_OPTS parameter allows the usage of auxiliary inputs with less than good status. The status of unused inputs is ignored.

The status of the output will be that of the PV, except for when the status of PV is good and the status of a used auxiliary input is not good and INPUT_OPTS is not configured to use it. In this case, the status of OUT will be Uncertain.

Schematic



PARAMETERS

Index	Parameter	DataType (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
1	ST_REV	Unsigned16		0	None	S	
2	TAG_DESC	OctString(32)		Spaces	Na	S	
3	STRATEGY	Unsigned16		0	None	S	
4	ALERT_KEY	Unsigned8	1 to 255	0	None	S	
5	MODE_BLK	DS-69		O/S	Na	S	See Mode Parameter
6	BLOCK_ERR	Bitstring(2)			E	D / RO	
7	PV	DS-65			PV	D / RO	Process analog value for use in executing the function.
8	OUT	DS-65	OUT_SCALE +/- 10%		OUT	D / Man	The analog value calculated as a result of executing the function.
9	PRE_OUT	DS-65			OUT	D / RO	Displays what would be the OUT value and status if the mode was Auto or lower.
10	PV_UNITS	Unsigned16		0	PV	S	The engineering units index for display. See Arithmetic block.
11	OUT_UNITS	Unsigned16		0	OUT	S	The engineering units of the output for display.

Index	Parameter	Data Type (length)	Valid Range/Options	Default Value	Units	Store / Mode	Description
12	GRANT_DENY	DS-70		0	na	D	Options for controlling access of host computer and local control panels to operating, tuning and alarm parameters of the block.
13	INPUT_OPTS	Bitstring(2)		0	na	S / O/S	Option bit string for handling the status of the auxiliary inputs.
14	IN	DS-65			PV	D	The primary input value of the block, required for blocks that filter the input to get the PV.
15	IN_LO	DS-65			PV	D	Input for the low range transmitter, in a range extension application.
16	IN_1	DS-65			None	D	Numbered input 1.
17	IN_2	DS-65			None	D	Numbered input 2.
18	IN_3	DS-65			None	D	Numbered input 3.
19	RANGE_HI	Float		0	PV	S	Constant value above which the range extension has switched to the high range transmitter.
20	RANGE_LO	Float		0	PV	S	Constant value below which the range extension has switched to the low range transmitter.
21	BIAS_IN_1	Float		0	None	S	The constant to be added to IN_1.
22	GAIN_IN_1	Float		0	None	S	The constant to be multiplied times (IN_1 + bias).
23	BIAS_IN_2	Float		0	None	S	The constant to be added to IN_2.
24	GAIN_IN_2	Float		0	None	S	The constant to be multiplied times (IN_2 + bias).
25	BIAS_IN_3	Float			None	S	The constant to be added to IN_3.
26	GAIN_IN_3	Float		0	None	S	The constant to be multiplied times (IN_3 + bias).
27	COMP_HI_LIM	Float		0	None	S	The high limit imposed on the PV compensation term.
28	COMP_LO_LIM	Float		0	None	S	The low limit imposed on the PV compensation term.

Index	Parameter	DataType (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
29	ARITH_TYPE	Unsigned8	1= Flow comp. linear 2= Flow comp. square root 3= Flow comp. approx. 4= BTU flow 5= Traditional mult. div. 6= Average 7= Traditional summer 8= Fourth order polynomial 9= HTG comp. level	0	E	S	The identification number of the arithmetic algorithm.
30	BAL_TIME	Float	Positive	0	Sec	S	This specifies the time for the internal working value of bias or ratio to return to the operator set bias or ratio, in seconds. In the PID block, it may be used to specify the time constant at which the integral term will move to obtain balance when the output is limited and the mode is Auto, Cas, or RCas.
31	BIAS	Float		0	OUT	S	The bias value used in computing the function block output, expressed in engineering units.
32	GAIN	Float		0	None	S	Dimensionless value used by the block algorithm in calculating the block output.
33	OUT_HI_LIM	Float		100	OUT	S	Limits the maximum output value.
34	OUT_LO_LIM	Float		0	OUT	S	Limits the minimum output value
35	UPDATE_EVT	DS-73			Na	D	This alert is generated by any change to the static data.

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
36	BLOCK_ALM	DS-72			Na	D	The block alarm is used for all configuration, hardware, connection failure or system problems in the block. The cause of the alert is entered in the subcode field. The first alert to become active will set the Active status in the Status attribute. As soon as the Unreported status is cleared by the alert reporting task, another block alert may be reported without clearing the Active status, if the subcode has changed.

Legend: *E* – Enumerated parameter; *na* – Adimensional parameter; *RO* – Read only; *D* – dynamic;
N – non-volatile; *S* – static

If BEHAVIOR parameter is “Adapted”:

The default value of ARITH_TYPE is the Gas flow compensation for linear transmitters, equation type 1.

The required mode for writing is the actual mode, regardless the target mode : OUT

Equation Types

ARITH_ TYPE	Equation
1 Flow comp. Linear	$OUT = PV * f * GAIN + BIAS$ where $f = \left[\frac{T1}{T2} \right]$ is limited
2 Flow comp. Square root	$OUT = PV * f * GAIN + BIAS$ where $f = \left[\sqrt{\frac{T1}{T2 * T3}} \right]$ is limited
3 Flow comp. Approx.	$OUT = PV * f * GAIN + BIAS$ where $f = \left[\sqrt{T1 * T2 * T3^2} \right]$ is limited
4 BTU flow	$OUT = PV * f * GAIN + BIAS$ where $f = [T1 - T2]$ is limited
5 Traditional mult. div.	$OUT = PV * f * GAIN + BIAS$ where $f = \left[\frac{T1}{T2} + T3 \right]$ is limited
6 Average	$OUT = \frac{PV + T1 + T2 + T3}{f} * GAIN + BIAS$ Where f is number of inputs used in computation (unusable inputs are not used).
7 Traditional summer	$OUT = (PV + T1 + T2 + T3) * GAIN + BIAS$
8 Fourth order polynomial	$OUT = (PV + T1^2 + T2^3 + T3^4) * GAIN + BIAS$
9 HTG comp. level	$OUT = \frac{PV - T1}{PV - T2} * GAIN + BIAS$

Examples

ARITH_T YPE	Example	Example equation	Note
1	Gas flow compensation for linear transmitters (e.g. turbine)	$Q_b = Q_f * K * \frac{P}{T}$	
2	Gas flow compensation For DP transmitters.	$Q_b = Q_f * K * \sqrt{\frac{P}{T * Z}}$	Z may be constant or an input from other block (AGA3)
3	Approx. liquid & steam flow comp.	$Q_b = Q_f * K * \sqrt{(K + K * T + K * T^2)}$ $Q_b = Q_f * K * \sqrt{(K + K * P)}$	Temperature connected to 3 and 4
4	BTU meter (heat flow)	$Q_{HEAT} = K * Q_{VOL} * (t_1 - t_2)$	
5	Simple "hard" (non-cascade) ratio	$Q_{SP} = Q_{WILD} * RATIO$	output is setpoint to PID block
6	Average of four temperature measurements	$t_a = \frac{t_1 + t_2 + t_3 + t_4}{f}$	.
7	Difference in pressure (or level)	$P_{bm} = P_b - P_m$	
9	Simple HTG compensated level	$h_{BT} = \frac{P_B - P_T}{P_B - P_M} * h_{BM}$	

NOTE: Square root of the third power may be achieved by selecting ARITH_TYPE = 3 and connecting input to IN and IN_1. Square root of the fifth power may likewise be achieved by connecting the input to IN, IN_1 and IN_3.

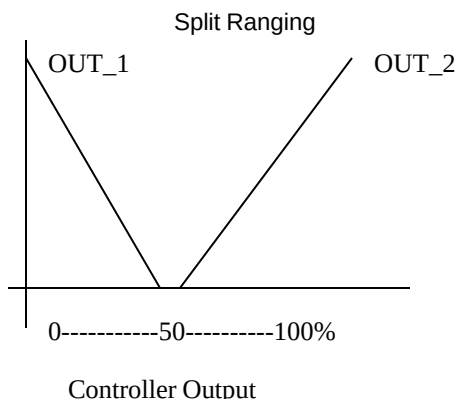
SPLT-Splitter

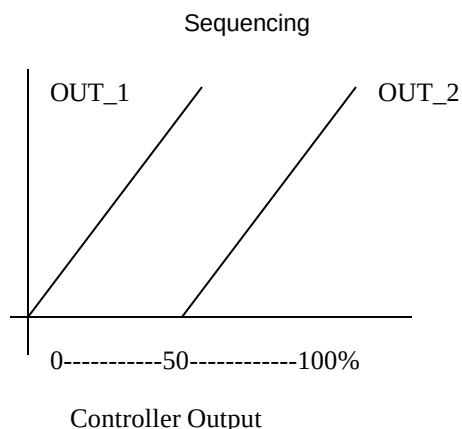
Description

The splitter block provides the capability to drive multiple outputs from a single input, usually a PID. This block would normally be used in split ranging or sequencing of multiple valve applications. Included in the block features are the capability to open valves as part of a predetermined schedule and leave open or closed a given valve after the controller has transitioned off the valve. The splitter supports two outputs. Since this block will participate in the control path after a PID block, back calculation support is included.

The application(s) targeted for the splitter block would be a single output from a controller which is used to control up to two valves in either split ranging fashion or as sequenced operation. Split ranging is that application where two valves are used such as a reactor where heating and cooling must be applied by the same controller. The controller action, direct or reversing, is implicitly reversed owing to the change in slope of the function as the input increases or decreases. Sequencing as applied to this document is when two or more valves are used to manipulate the flow of some material and the controller action is not reversed implicitly or otherwise. An example is pH control where added valves are required to increase the loop rangeability.

A graphical representation of the splitter vs. controller output is shown below. Both a Split range and Sequencing application are shown.





The following parameters are used to specify the signal splitter operation:

X11, Y11 X12, Y12

X21, Y21 X22, Y22

Where X_{n1} is the value of CAS_IN associated with OUT_n and X_{n1} and X_{n2} refers to the 1st and 2nd coordinates respectively. The Y values are defined in the same way.

IN_ARRAY	OUT_ARRAY
X ₁₁	Y ₁₁
X ₁₂	Y ₁₂
X ₂₁	Y ₂₁
X ₂₂	Y ₂₂

By specifying the coordinates as shown above the endpoints of the lines are defined. The contents of the respective X's and Y's are held in IN_ARRAY and OUT_ARRAY parameters. If a set of points are specified such that a region of the input range is not specified then the block will interpolate to the endpoint of the input value, either high or low.

The parameter LOCKVAL provides an option to specify whether OUT_1 remains at its ending level when control is switched to OUT_2, or goes to zero. If LOCKVAL is true, OUT_1 remains at its current value when OUT_2 is non-zero. If LOCKVAL is false then as soon as the OUT_2 becomes non-zero then OUT_1 goes to zero.

Supported Modes

O/S, IMAN, Auto and Cas.

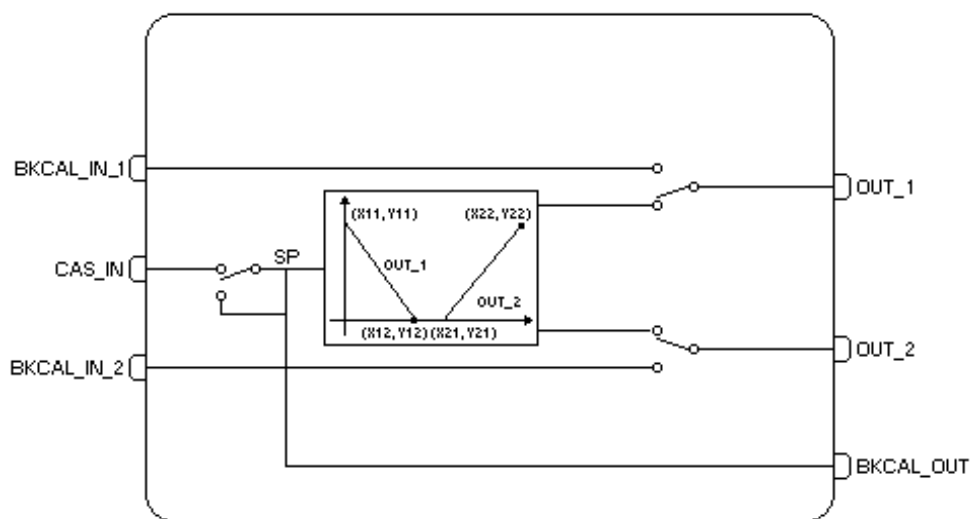
Status Handling

Sub-status values received at CAS_IN will be passed to both outputs, except for those used in the cascade handshake. **An IFS will go to both the active and the inactive outputs.** The back calculation status will only come from the active output. An output held by LOCKVAL is not active. Limit status must be inverted if the slope of the active output is negative. No limits are sent back on BKCAL_OUT if neither output is active.

Actual mode	BKCAL_IN_1	BKCAL_IN_2	BKCAL_OUT	ACTION
Iman	NI	NI	NI	BKCAL_OUT limited high and low to $(X12 + X21)/2$
Auto or Cas	NI	OK	OK	BKCAL_OUT limited to X21 low and X22 high
Auto or Cas	OK	NI	OK	BKCAL_OUT limited to X11 low and X12 high
Iman	IR	NI	IR	Initialize cascade to value given by curve X1 vs Y1
Auto or Cas	IR	OK	OK	Initialize OUT_1 using internal offset from Y1
Iman	NI	IR	IR	Initialize cascade to value given by curve X2 vs Y2
Auto or Cas	OK	IR	OK	Initialize OUT_2 using internal offset from Y2

Legend: NI-not invited; IR-initialization request; OK-working in cascade

Schematic



PARAMETERS

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
1	ST_REV	Unsigned16		0	None	S	
2	TAG_DESC	OctString(32)		Spaces	Na	S	
3	STRATEGY	Unsigned16		0	None	S	
4	ALERT_KEY	Unsigned8	1 to 255	0	None	S	
5	MODE_BLK	DS-69		O/S	Na	S	See Mode Parameter
6	BLOCK_ERR	Bitstring(2)			E	D	
7	SP	DS-65				N / Auto	The analog set point.
8	OUT_1	DS-65			OUT1	D / RO	Numbered output parameter 1.
9	OUT_2	DS-65			OUT2	D / RO	Numbered output parameter 2.
10	OUT_1_UNITS	Unsigned16		0	E	S	The units code for the corresponding output.
11	OUT_2_UNITS	Unsigned16		0	E	S	The units code for the corresponding output.
12	GRANT_DENY	DS-70		0	Na	S / O/S	Options for controlling access of host computer and local control panels to operating, tuning and alarm parameters of the block.
13	STATUS_OPTS	Bitstring(2)		0	Na	S	Options which the user may select in the block processing of status.
14	CAS_IN	DS-65				D	This parameter is the remote setpoint value, which must come from another Fieldbus block, or a DCS block through a defined link.
15	BKCAL_OUT	DS-65				D / RO	The value and status required by an upper block's BKCAL_IN so that the upper block may prevent reset windup and provide bumpless transfer to closed loop control.
16	IN_ARRAY	4 Floats		0's		S	An array which contains the values of the input or X variables.
17	OUT_ARRAY	4 Floats		0's		S	An array which contains the values of the output or Y variables.
18	LOCKVAL	Unsigned8	0:FALSE 1:TRUE	FALSE	E	S	Flag for holding the first output at current value when the other output is non-zero.
19	BKCAL_IN_1	DS-65				N	The back calculated input required to initialize a lower cascade 1.
20	BKCAL_IN_2	DS-65				N	The back calculated input required to initialize a lower cascade 2.
21	BAL_TIME	Float		0	Sec	S	This specifies the time for the internal working value of bias or ratio to return to the operator set bias or ratio, in seconds.
22	UPDATE_EVT	DS-73			na	D	This alert is generated by any change to the static data.

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
23	BLOCK_ALM	DS-72			na	D	The block alarm is used for all configuration, hardware, connection failure or system problems in the block. The cause of the alert is entered in the subcode field. The first alert to become active will set the Active status in the Status attribute. As soon as the Unreported status is cleared by the alert reporting task, another block alert may be reported without clearing the Active status, if the subcode has changed.

Legend: *E – Enumerated parameter; na – Adimensional parameter; RO – Read only; D – dynamic; N – non-volatile; S - static*

If BEHAVIOR parameter is “Adapted”:

The required mode for writing is the actual mode, regardless the target mode : SP

CHAR - Signal Characterizer

Description

The signal characterizer block has two sections, each with an output that is a non-linear function of the respective inputs. The function is determined by a single look-up table with x-y coordinates of twenty points each. The status of input is copied to the corresponding output, so the block may be used in the control or process signal path. An option can swap the axes of the function for section 2, so that it can be used in the backward control path.

The block interpolates correlates the input IN_1 to the output OUT_1 and the input IN_2 to the output OUT_2

according to a curve given by the points:

$[X_1; Y_1], [X_2; Y_2], \dots, [X_{20}; Y_{20}]$

where x corresponds to the Input and y to the Output. The x-coordinates are given in engineering units of the input. The y-coordinates are given in engineering units of the output.

How to configure not used points

If the curve has m points, $m < 20$, the non-configured points, $[X_{m+1}; Y_{m+1}], [X_{m+2}; Y_{m+2}], \dots, [X_{20}; Y_{20}]$ shall be set with +INFINITY.

BLOCK_ERR

The BLOCK_ERR will reflect the following causes:

Block Configuration Error – the configuration error occurs when the BYPASS parameter has an invalid value or the curve has any of the following problems:

$CURVE\_X[i] > CURVE\_X[i+1]$

If the curve is not using **effectively** 20 points and any non-configured point is different from +INFINITY.

If SWAP_2 is true and the curve is not monotonic.

Out of Service – it occurs when the block is in O/S mode.

Supported Modes

O/S and AUTO.

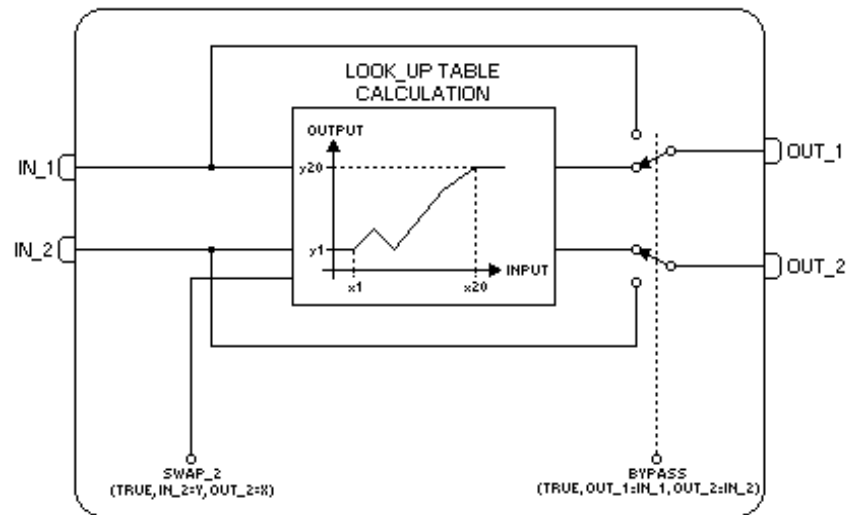
Status Handling

The quality and sub-status of OUT_1 and OUT_2 reflect the status of IN_1 and IN_2, respectively.

If one of the curve limits is reached, the appropriate limit is indicated. Limits are reversed if the curve slope is negative.

The status of output will be Bad – Configuration Error if there is an error as indicated in the BLOCK_ERR parameter.

Schematic



Parameters

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
1	ST_REV	Unsigned16		0	None	S	
2	TAG_DESC	OctString(32)		Spaces	Na	S	
3	STRATEGY	Unsigned16		0	None	S	
4	ALERT_KEY	Unsigned8	1 to 255	0	None	S	
5	MODE_BLK	DS-69		O/S	Na	S	See Mode Parameter
6	BLOCK_ERR	Bitstring(2)			E	D	
7	OUT1	DS-65			Y	D / RO	Numbered output parameter 1. The interpolation result of IN_1.
8	OUT2	DS-65			X or Y	D / RO	Numbered output parameter 2. The interpolation result of IN_2.
9	X_UNITS	Unsigned16			E	S	The engineering unit of the variables corresponding to the x-axis for display.
10	Y_UNITS	Unsigned16			E	S	The engineering unit of the variables corresponding to the y-axis for display.
11	GRANT_DENY	DS-70		0	Na	D	Options for controlling access of host computer and local control panels to operating, tuning and alarm parameters of the block.
12	CONTROL_OPT S	Bitstring(2)	See Block Options	0	Na	S / O/S	See Block Options
13	IN_1	DS-65				D	Numbered input parameter 1.
14	IN_2	DS-65				D	Numbered input parameter 2.

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
15	BYPASS	Unsigned8	1:Off 2:On	0	E	S / Man	When bypass is set, the input value will be directly transferred to the output.
16	SWAP_2	Unsigned8	True: OUT_2 is the x axis and IN_2 the y axis.	FALSE	E	S	Changes the algorithm in such a way that IN_2 corresponds to "y" and OUT_2 to "x".
17	CURVE_X	20 Floats		0's	X	S	Curve input points. The x_i points of the curve are defined by an array of twenty points.
18	CURVE_Y	20 Floats		0's	Y	S	Curve output points. The y_i points of the curve are defined by an array of twenty points.
19	UPDATE_EVT	DS-73			na	D	This alert is generated by any change to the static data.
20	BLOCK_ALM	DS-72			na	D	The block alarm is used for all configuration, hardware, connection failure or system problems in the block. The cause of the alert is entered in the subcode field. The first alert to become active will set the Active status in the Status attribute. As soon as the Unreported status is cleared by the alert reporting task, another block alert may be reported without clearing the Active status, if the subcode has changed.

Legend: E – Enumerated parameter; na – Adimensional parameter; RO – Read only; D – dynamic; N – non-volatile; S - static

If BEHAVIOR parameter is "Adapted":

The default value of BYPASS is Off.
The default value of all elements of CURVE_X is +INF.
The default value of all elements of CURVE_Y is +INF.

INTG - Integrator

Description

The Integrator Function Block integrates a variable in function of the time or accumulates the counting of a Pulse Input block. The integrated or accumulated value is compared to pre-trip and trip limits, generating discrete signals when these limits are reached. The integrated value may go up, starting from zero, or down, starting from the trip value (parameter SP). In order to determine the amount of uncertain or bad readings, the block integrates the variables with bad or bad and uncertain status separately (parameter RTOTAL). The values used in this second integration are the values with good status just before they went from good to bad or uncertain. Featuring two flow inputs, the block can calculate and integrate net flow. This can be used to calculate volume or mass variation in vessels or as an optimizing tool for flow ratio control.

The basic function of the Integrator block is to integrate an analog value over time. It can also accumulate the count pulses coming from Pulse Input blocks or from another Integrator Block. This block is normally used to totalize flow, giving total mass or volume over a certain time, or totalize power, giving the total energy. The algorithm shall comply with the following points:

1) The block has two dual purpose inputs, IN_1 and IN_2. Each input can receive a measurement per unit of time (rate) or an accumulated number of pulses. Each input can receive one of the following types of variables:

RATE - When the variable connected to the input is a rate, i.e., kg/s, W, Gal/hour, etc. This input can come from, or is derived from the output OUT of a Pulse Input block or from the output of an Analog Input block.

ACCUM - When the input comes from the OUT_ACCUM of a Pulse Input block, which represents a continuous accumulation of pulse counts from a transducer.

2) The input type is configured in the bit string parameter INTEG_OPTS. The bits corresponding to IN_1 and IN_2 can be set as follows:

false = RATE
true = ACCUM

3) Rate inputs:

- Each input needs a parameter to define the rate time unit: [TIME_UNIT1] or [TIME_UNIT2]. The time units are used to convert the two rates in units of mass, volume or energy per second.
- The second analog input may have to be converted into the same units of the first input. This is achieved by a unit conversion factor, given by the parameter [UNIT_CONV].

Each rate, multiplied by the block execution time, gives the mass, volume or energy increment per block execution. This increment will be added or subtracted in a register, according to some rules defined ahead.

Counter inputs:

- The counter inputs are originated in Pulse Input blocks. It can also be connected to any floating point variable, as the output of another integrator block. The output ACCUM_OUT of the Pulse Input block represents a continuous accumulation of pulses from the flow transducer, while the output OUT of an Integrator represents an integration or accumulation of analog inputs.

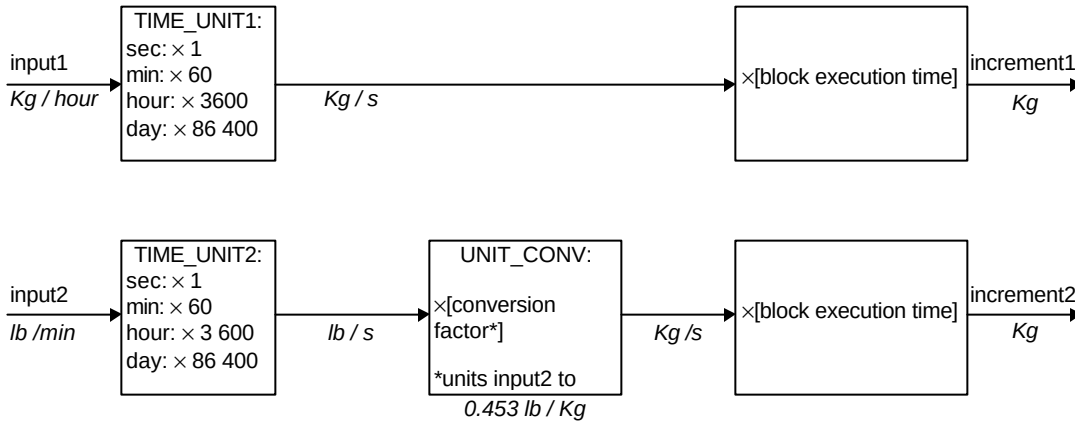


Fig. 1 - Increment calculation with rate input

- The Integrator block shall determine the variation of the counter input readings since the last execution.
- As the output ACCUM_OUT of the Pulse Input block wraps up when the counting reaches 999,999 and does not increment or decrement by more than 499,999 per cycle, the variation is determined as follows:

a) If the difference between the reading in one cycle and the reading in the preceding cycle is smaller than 500,000 or larger than (-500,000), the difference will be taken as variation.

• If the difference between the reading in one cycle and the reading in the preceding cycle is larger or equal to (+500,000), add (-1.000,000) and use the result as the variation.

c) If the difference between the reading in one cycle and the reading in the preceding cycle is smaller or equal to (-500,000), add (+1.000,000) and use the result as the variation.

• If the output OUT of another integrator block is used, that block should be programmed to have only increasing counting.

• The variation of each input must be multiplied by the value, in engineering units, of each pulse given by:

PULSE_VAL1 and PULSE_VAL2.

The result is the increment in engineering units of, for example, mass, volume or energy per block execution.

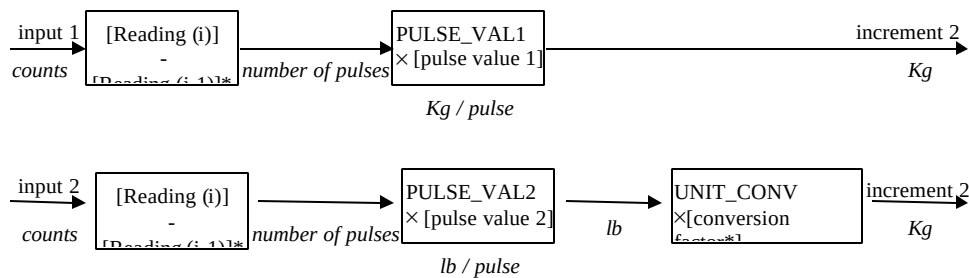


Fig. 2- Increment calculation with counter input

5) In order to distinguish forward and reverse flows, the Integrator block considers a negative sign as an indication of reverse flow. Some flow meters already indicate forward and reverse flows by adding a sign to the measurement value. Others use a separate binary signal.

This signal can be connected to the inputs REV_FLOW1 and REV_FLOW2, with the following options:

0 = False (not invert signal)
1 = True (invert signal)

where True will invert the signal of the IN_i signal.

6) The net flow is obtained by adding the two increments. The net increment will have a positive or negative signal to indicate the net flow direction. In order to integrate the difference between the inflow and outflow of a tank, for example, the second one can be assigned to be negative.

7) The net flow direction to be considered in the totalization is defined in INTEG_OPTS. The following options are available:

FORWARD = only positive flows (after application of REV_FLOW_i) are totalized. The negative values will be treated as zero. FORWARD is selected when the bit corresponding to Forward is set to *true*.

REVERSE = only negative flows are totalized. The positive values will be treated as zero. The option bit Reverse must be set to *true*.

TOTAL = both positive and negative values will be totalized. Both option bits Forward and Reverse must be set to *true* or to *false*.

8) There are two values for the totalization:

TOTAL (OUT) = Where the increments are added every cycle when the inputs have the status *good*, *good* or *uncertain*, or last usable value if *bad*. The status to be considered as usable are defined in INTEG_OPTS.

RTOTAL = The rejected increments with status not selected by INTEG_OPTS are added according to some rules described below.

9) The main totalization TOTAL may use inputs with different status, according to a selection established by INTEG_OPTS:

USE_UNCERTAIN = Use *good* and *uncertain* inputs when this bit is set to *true*.

USE_BAD = Use the last *good* or *uncertain* input before it went to *bad* when this bit is set to *true*.

USE_GOOD = Use *good* inputs when both former bits are set to *false*.

If both bits are set to *true*, the option USE_ANY is selected.

10) The increments with status not selected by INTEG_OPTS are totalized in RTOTAL, according to the following rules:

a) If INTEG_OPTS has no selection for status, the increments with *uncertain* or *bad* status must be added to RTOTAL. The last value with good status is used in place of the input value with *bad* status.

b) If INTEG_OPTS has the USE_UNCERTAIN bit set, only the increments originated by inputs with *bad* input status shall be added to RTOTAL. The last usable value is used in the totalization.

- If INTEG_OPTS has the USE_BAD bit set, the last usable value goes to TOTAL and nothing to RTOTAL..

11) TOTAL can be read in the output OUT. The engineering units used in the indication is defined in OUT_UNITS.

12) The block has a discrete input to reset the integration - RESET_IN. While this input parameter has value TRUE, the block will be old reset, therefore it will start to integrate only after it goes to FALSE.

The operator can send an operator command to reset the counting by making OP_CMD_INT = RESET. The mechanism to reset using this parameter is different from that one used in the RESET_IN, because a write operation in the OP_CMD parameter with RESET value will cause a reset, but the block will start to integrate soon after it, despite of the OP_CMD_INT remains with RESET.

The integration can start from zero and go up or it can start from a Setpoint value (SP) and go down. The totalization may be reset in different ways. This is defined by the parameter INTEG_TYPE:

UP_AUTO Counts up starting from zero with automatic reset when SP is reached

UP_DEM Counts up starting from zero with demand reset

DN_AUTO Counts down starting from SP with auto reset when zero is reached

DN_DEM Counts down starting from SP with demand reset

PERIODIC: Counts up and is reset periodically according to CLOCK_PER

DEMAND: Counts up and is reset on demand

PER&DEM Counts up and is reset periodically or on demand

15)Block takes a snapshot of TOTAL(OUT), RTOTAL and SP prior the reset and keep the information in the registers STOTAL, SRTOTAL and SSP, respectively. The information is kept at least until the next reset.

16)The integrator shall reject reset requests for 5 seconds after a reset. This is to guarantee that the snapshots values are exposed to FIELD BUS before they can be overwritten.

17)When the integration is counting up and it reaches a limit given by the parameter Setpoint - SP - , the discrete output OUT_TRIP is set.

When the integration is counting down, it starts from a value given by SP. When the counting reaches zero, the discrete output OUT_TRIP is set.

18)When the integration is counting up and it reaches a limit given by:

SP - [PRE_TRIP]

the discrete output OUT_PTRIP is set.

When the integration is counting down and it reaches [PRE_TRIP], the discrete output OUT_PTRIP is set.

OUT_PTRIP is set until SP or zero is reached.

When the options UP_AUTO or DN_AUTO are selected, a residue beyond the trip value may be considered in the next batch if the option CARRY, of INTEG_OPTS is set. This residue will be "carried" to the next batch by:

- a) Start counting from the residue value, instead of zero, when UP_AUTO is selected.
- b) Start counting from (SP-Residue) when DN_AUTO is selected.

20)The number of resets is counted in the register N_RESET. This counter can not be written or reset.

21)The percentage of rejected counts (RTOTAL) in the whole totalization (TOTAL+RTOTAL) may be determined by calculating the parameter PCT_INCL as it follows:

$$PCT\_INCL = TOTAL / (TOTAL+RTOTAL)$$

The acceptable limit for *good* status is established by GOOD_LIM.

The acceptable limit for *uncertain* status is established by UNCERT_LIM. If

$PCT\_INCL \geq GOOD\_LIM$, the status of OUT will be *good*,
else, if $PCT\_INCL \geq UNCERT\_LIM$ the status will be *uncertain*,
else, the status will be *bad*.

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
							of integration.
9	OUT_UNITS	Unsigned16		0	OUT	S	The engineering units of the output for display.
10	GRANT_DENY	DS-70		0	Na	D	Options for controlling access of host computer and local control panels to operating, tuning and alarm parameters of the block.
11	IN_1	DS-65		0		D	Input parameter for flow 1.

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
12	IN_2	DS-65		0		D	Input parameter for flow 2.
13	OUT_TRIP	DS-66		0	On/Off	D	Output parameter for trip indication.
14	OUT_PTRIP	DS-66		0	On/Off	D	Output parameter for pre-trip indication.
15	TIME_UNIT1	Unsigned8	1:seconds 2:minutes 3:hours 4:days	0	E	S	Converts the rate time units in seconds.
16	TIME_UNIT2	Unsigned8	1:seconds 2:minutes 3:hours 4:days	0	E	S	Converts the rate time units in seconds.
17	UNIT_CONV	Float		1	none	S	Factor to convert the engineering units of input 2 into the engineering units of input 1.
18	PULSE_VAL1	Float		0	none	S	Determines the mass, volume or energy per pulse. It is used only in accum mode.
19	PULSE_VAL2	Float		0	none	S	Determines the mass, volume or energy per pulse. It is used only in accum mode.
20	REV_FLOW1	DS-66	0=FORWARD 1=REVERSE		E	D	It indicates reverse flow in IN_1 when <i>true</i> .
21	REV_FLOW2	DS-66	0=FORWARD 1=REVERSE		E	D	It indicates reverse flow in IN_2 when <i>true</i> .
22	RESET_IN	DS-66	0:Off 1:Reset		E	D	Resets the totalizer - OUT parameter.
23	STOTAL	Float		0	OUT	D	Indicates the snapshot of the totalizer - OUT parameter just before a reset.
24	RTOTAL	Float		0	OUT	D	Indicates the totalization of <i>bad</i> or <i>bad</i> and <i>uncertain</i> inputs, according to INTEG_OPTS.
25	SRTOTAL	Float		0	OUT	D	The snapshot of RTOTAL just before a reset.
26	SSP	Float		0	OUT	D	The snapshot of SP.
27	INTEG_TYPE	Unsigned8	1=UP_AUTO 2=UP_DEM 3=DN_AUTO 4=DN_DEM 5=PERIODIC 6=DEMAND 7=PER&DEM	0	E	S	Defines the type of counting (up or down) and the type of resetting (demand or periodic)
28	INTEG_OPTS	Bitstring(2)			none	S	A bit string to configure the type of input (rate or accum.) used in each input, the flow direction to be considered in the totalization, the status to be considered in TOTAL and if the totalization residue shall be used in the next batch (only when INTEG_TYPE = UP_AUTO or DN_AUTO).

Index	Parameter	Data Type (length)	Valid Range/Options	Default Value	Units	Store / Mode	Description
29	CLOCK_PER	Float		0	Sec	S	Establishes the period for periodic reset, in hours.
30	PRE_TRIP	Float		0	OUT	S	Adjusts the amount of mass, volume or energy that will set OUT_PTRIP when the integration reaches (SP-PRE_TRIP) when counting up or PRE_TRIP when counting down.
31	N_RESET	DS-65			none	N / RO	Counts the number of resets. It can not be written on nor reset.
32	PCT_INCL	Float			%	D / RO	Indicates the percentage of inputs with <i>good</i> stati compared to the ones with <i>bad</i> or <i>uncertain</i> and <i>bad</i> stati.
33	GOOD_LIM	Float	0 to 100%	0.1	%	S	Sets the limit for PCT_INCL. Below this limit OUT receives the status <i>good</i> .
34	UNCERT_LIM	Float	0 to 100%	0.2	%	S	Sets the limit for PCT_INCL. Below this limit OUT receives the status <i>uncertain</i> .
35	OP_CMD_INT	Unsigned8	0=Undefined 1=Reset		E	D	Operator command. RESET. Resets the totalizer.
36	OUTAGE_LIM	Float	Positive	0	Sec	S	The maximum tolerated duration for power failure. This feature is not supported.
37	UPDATE_EVT	DS-73			na	D / RO	This alert is generated by any change to the static data.
38	BLOCK_ALM	DS-72			na	D	The block alarm is used for all configuration, hardware, connection failure or system problems in the block. The cause of the alert is entered in the subcode field. The first alert to become active will set the Active status in the Status attribute. As soon as the Unreported status is cleared by the alert reporting task, another block alert may be reported without clearing the Active status, if the subcode has changed.

Legend: *E* – Enumerated parameter; *na* – Adimensional parameter; *RO* – Read only; *D* – dynamic; *N* – non-volatile; *S* - static

If BEHAVIOR parameter is “Adapted”:

The default value of TIME_UNIT1 is seconds.
The default value of TIME_UNIT2 is seconds.
The default value of INTEG_TYPE is UP_DEM.
The required mode for writing is the actual mode, regardless the target mode : SP and OUT

AALM - Analog Alarm

Description

The Analog Alarm Block provides alarm condition reporting on an analog output of any block. Alarm conditions include high, high-high, low, and low-low alarms. These limits are computed based on gains and biases from a process setpoint input, thus providing dynamic deviation alarming. An option to temporarily expand alarm limits after a setpoint change is provided. Also, an alarm condition may be ignored for a specified period of time to avoid nuisance alarm reporting.

The input value, IN, is filtered according to the PV_FTIME time-constant, to become PV. PV is then alarmed in the *auto* mode.

Alarm limits may be dynamically calculated from process setpoint (PSP). The operating limits (same parameter names as limits, but suffixed with an "X") are calculated from specified gains and biases as follows:

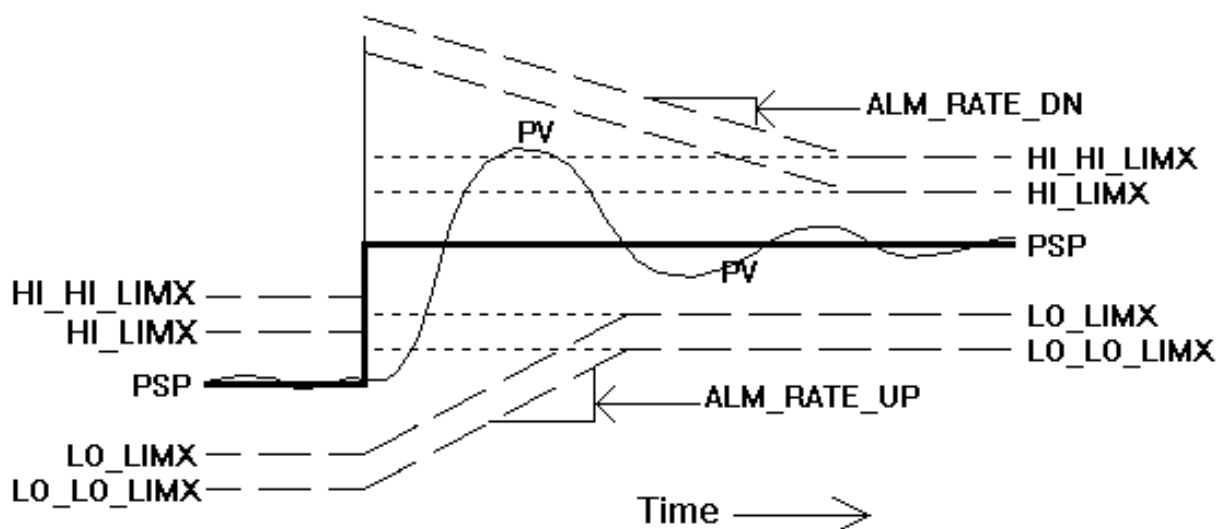
$HI\_HI\_LIMX = PSP * HI\_GAIN + HI\_HI\_BIAS + EXPAND\_UP$ (or default to HI_HI_LIM if any used parameter is undefined)

$HI\_LIMX = PSP * HI\_GAIN + HI\_BIAS + EXPAND\_UP$ (or default to HI_LIM if any used parameter is undefined)

$LO\_LIMX = PSP * LO\_GAIN - LO\_BIAS - EXPAND\_DN$ (or default to LO_LIM if any used parameter is undefined)

$LO\_LO\_LIMX = PSP * LO\_GAIN - LO\_LO\_BIAS - EXPAND\_DN$ (or default to LO_LO_LIM if any used parameter is undefined)

Effective alarm limits may be temporarily expanded on step setpoint changes to avoid nuisance alarms. The operating high alarm limits are increased by a calculated term, EXPAND_UP. The operating low alarm limits are decreased by a calculated term, EXPAND_DN. See the example in the following time chart:



Both the level 1 (advisory) and level 2 (critical) effective alarm limits are expanded after a setpoint change by the absolute value of the change to PSP. The expansions then decay toward the base limits at a rate determined by ALM_RATE_UP and ALM_RATE_DN parameters. This permits normal and over-damped process responses to avoid alarms on the initial change and permits under-damped process responses to avoid alarms on overshooting or ringing. The following properties and rules apply:

- The four limits initially expand by the same value, the setpoint change.

The two high limits always expand by the same value, EXPAND_UP, and decay by the same rate, ALM_RATE_DN (which may differ from the low limits).

- The two low limits always expand by the same value, EXPAND_DN, and decay by the same rate, ALM_RATE_UP (which may differ from the high limits).

- The expansion feature may be suppressed in the upward direction by setting ALM_RATE_DN to zero. The expansion feature may be suppressed in the downward direction by setting ALM_RATE_UP to zero.

- Additional step setpoint changes prior to complete decay of a previous expansion will expand the alarm limits in each direction to the maximum of the remaining expansion value or new expansion value.

The existence of a new alarm condition may be temporarily ignored by setting the IGNORE_TIME parameter to the number of seconds to disregard the alarm. Both the reporting of the alarm and the possible change to PRE_OUT_ALM will be ignored during this time. This parameter does not delay the clearing of the existence of the alarm on return-to-normal. If the alarm condition does not persist for IGNORE_TIME seconds, it will not be reported.

PRE_OUT_ALM and OUT_ALM indicate the existence of one or more selected alarm conditions per the specification of the OUT_ALM_SUM parameter. Enumerated choices of the OUT_ALM_SUM parameter and their included alarm conditions are listed below:

OUT_ALM_ SUM	INCLUDED ALARM CONDITIONS			
	HI_HI_ ALM	HI_ ALM	LO_ ALM	LO_LO_ ALM
ANY	✓	✓	✓	✓
LOWs			✓	✓
HIGHs	✓	✓		
LEVEL1		✓	✓	
LEVEL2	✓			✓
LO_LO				✓
LO			✓	
HI		✓		
HI_HI	✓			
NONE				

For example, if LOWs is chosen for OUT_ALM_SUM, either a LO_ALM or LO_LO_ALM being *true* will cause OUT_ALM to be set to *true*. If LEVEL1 is chosen for OUT_ALM_SUM, either a LO_ALM or HI_ALM being *true* will cause OUT_ALM to be set to *true*.

The OUT_ALM parameter can be used for control purposes, for example , as an interlock signal, besides the basic function of alarm monitoring.

Simple alarm calculation: static alarm limits, no expansion and no delay to detection

The alarm limits will be static (HI_HI_LIM, HI_LIM, LO_LIM and LO_LO_LIM are the effective operating alarm limits) if the corresponding gain or bias is +/- INF, or the input PSP is left unconnected with status Bad – O/S.

The alarm limit expansion will be disabled by setting ALM_RATE_DN and ALM_RATE_UP to zero,.

The detection of an alarm will be without delay setting IGNORE_TIME to zero.

BLOCK_ERR

The BLOCK_ERR of the Arithmetic block will reflect the following causes:
Out of Service – When the block is in O/S mode.

Supported Modes

O/S, MAN and AUTO.

Status Handling

The block will not filter an IN value with a bad status or uncertain status (and "Use Uncertain" option in STATUS_OPTS is not set), but will instead, filter to the last usable value of PV and assign it the unusable status from IN. When the status of IN returns to a usable value (good or uncertain [and "Use Uncertain" option in STATUS_OPTS is set]), the value of PV will again be filtered toward the value of IN with the status of IN.

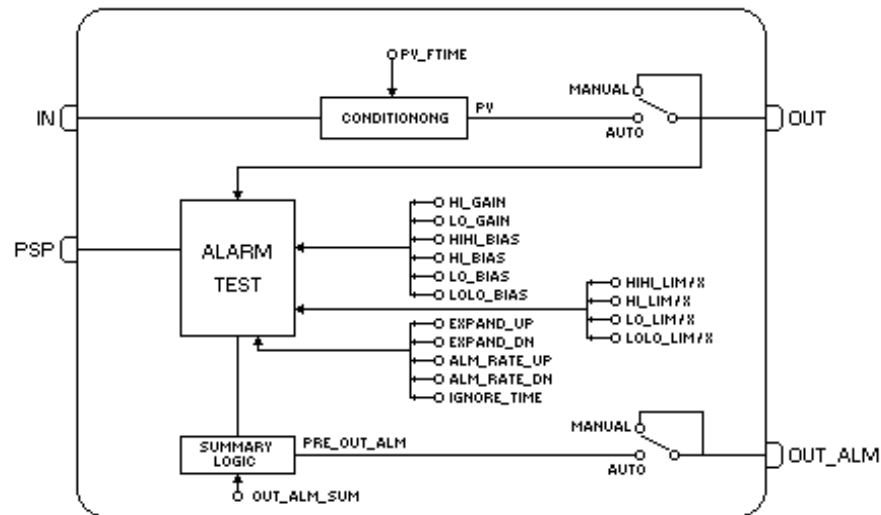
The status of OUT is set to the status of PV (and IN) when in auto mode.

If the worst quality of the stati of PV and PSP is bad, or uncertain (and "Use Uncertain" option in STATUS_OPTS is not set) the alarm test will not be performed and the status of PRE_OUT_ALM will be set to bad (non-specific). Otherwise, the alarm test will be performed and the quality of the

status of PRE_OUT_ALM will be set to the worst quality of the stati of PV and PSP (good or uncertain). While the alarm condition is not being evaluated due to unusable stati, existing alarms will not be cleared and new alarms will not be generated. Prior alarm conditions may still be acknowledged.

In auto mode, the status of OUT_ALM will be set to the status of PRE_OUT_ALM. In man mode, the limits status of OUT_ALM is set to double-limited.

Schematic



Parameters

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
1	ST_REV	Unsigned16		0	None	S	
2	TAG_DESC	OctString(32)		Spaces	Na	S	
3	STRATEGY	Unsigned16		0	None	S	
4	ALERT_KEY	Unsigned8	1 to 255	0	None	S	
5	MODE_BLK	DS-69		O/S	Na	S	See Mode Parameter
6	BLOCK_ERR	Bitstring(2)			E	D / RO	
7	PV	DS-65			PV	D / RO	Process analog value. This is the IN value after pass over the PV filter.
8	OUT	DS-65	OUT_SCALE +/- 10%		OUT	N / Man	The output value result of the block calculation.
9	OUT_SCALE	DS-68		0-100%	OUT	S / Man	The high and low scale values to the OUT parameter.
10	GRANT_DENY	DS-70		0	na	D	Options for controlling access of host computer and local control panels to operating, tuning and alarm parameters of the block.
11	STATUS_OPTS	Bitstring(2)	See Block Options	0	Na	S / O/S	See Block Options

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
12	PV_FTIME	Float	Non-Negative	0	Sec	S	Time constant of a single exponential filter for the PV, in seconds.
13	IN	DS-65			PV	D	The primary input value of the block, or PV value.
14	PSP	DS-65			OUT	D	This is the process setpoint, which may be used for alarm limit determination.
15	HI_GAIN	Float		1.1		S	This gain multiplies PSP before addition of biases for HI_LIM and HI_HI_LIM.
16	LO_GAIN	Float		0.9	PV/Sec	S	This gain multiplies PSP before subtraction of biases for LO_LIM and LO_LO_LIM.
17	HI_HI_BIAS	Float	Positive	1.0	PV/Sec	S	This bias is added to PSP*HI_GAIN to determine HI_HI_LIM.
18	HI_BIAS	Float	Positive	0.0	PV	S	This bias is added to PSP*HI_GAIN to determine HI_LIM.
19	LO_BIAS	Float	Positive	0.0	PV	S	This bias is subtracted from PSP*LO_GAIN to determine LO_LIM.
20	LO_LO_BIAS	Float	Positive	1.0	None	S	This bias is subtracted from PSP*LO_GAIN to determine LO_LO_LIM.
21	PRE_OUT_ALM	DS-66			E	D	This parameter is the alarm summary variable of the analog alarm block.
22	OUT_ALM	DS-66			E	D	This parameter is the alarm summary variable of the analog alarm block when in <i>Auto</i> mode and is the value specified by the operator/engineer in <i>Man</i> mode.
23	OUT_ALM_SUM	Unsigned8	0:NONE 1:LO_LO 2:LO 3:LOWs 4:HI 6:LEVEL1 8:HI_HI 9:LEVEL2 12:HIGHS 15:ANY	0	E	S	Specifies the alarms conditions which must be <i>true</i> in order for OUT_ALM to be set to <i>true</i> : ANY, LOWs, HIGHs, LEVEL1, LEVEL2, LO_LO, LO, HI, or HI_HI.

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
24	ALM_RATE_UP	Float	Positive	0.0	OUT/sec	S	Ramp rate at which downward alarm expansion due to step PSP changes is decayed in the upward direction. Expressed in engineering units per second. Expansion feature disabled in the downward direction if set to zero. (Positive)
25	ALM_RATE_DN	Float	Positive	0.0	OUT/sec	S	Ramp rate at which upward alarm expansion due to step PSP changes is decayed in the downward direction. Expressed in engineering units per second. Expansion feature disabled in the upward direction if set to zero. (Positive)
26	EXPAND_UP	Float			OUT	D	Amount, in engineering units, that base HI and HI_HI limits are expanded after a setpoint change. Dynamically calculated by block. Initially expanded by the amount of a setpoint change and decayed at the rate of ALM_RATE_UP. (Positive)
27	EXPAND_DN	Float			OUT	D	Amount, in engineering units, that base LO and LO_LO limits are expanded after a setpoint change. Dynamically calculated by block. Initially expanded by the amount of a setpoint change and decayed at the rate of ALM_RATE_DN. (Positive)
28	IGNORE_TIME	Float	Positive	0.0	Sec	S	The time, in seconds, to ignore the existence of a new alarm condition. There is no delay on clearing the existence of the alarm on return to normal. If the alarm does not persist for IGNORE_TIME seconds, it will not be reported. Does not apply to self-clearing (transient) type alarms.
29	UPDATE_EVT	DS-73			Na	D	This alert is generated by any change to the static data.

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
30	BLOCK_ALM	DS-72			Na	D	The block alarm is used for all configuration, hardware, connection failure or system problems in the block. The cause of the alert is entered in the subcode field. The first alert to become active will set the Active status in the Status attribute. As soon as the Unreported status is cleared by the alert reporting task, another block alert may be reported without clearing the Active status, if the subcode has changed.
31	ALARM_SUM	DS-74	See Block Options		Na	S	The current alert status, unacknowledged states, unreported states, and disabled states of the alarms associated with the function block.
32	ACK_OPTION	Bitstring(2)	0: Auto ACK Disable 1: Auto ACK Enable	0	Na	S	Selection of whether alarms associated with the block will be automatically acknowledged
33	ALARM_HYS	Float	0 to 50 %	0.5%	%	S	Alarm hysteresis parameter. In order to clear the alarm the amount the PV must return within the alarm limit plus hysteresis.
34	HI_HI_PRI	Unsigned8	0 to 15	0		S	Priority of the high high alarm.
35	HI_HI_LIM	Float	OUT_SCALE, +INF	+INF	PV	S	The setting for high high alarm in engineering units.
36	HI_HI_LIMX	Float	OUT_SCALE, +INF	+INF	PV	S	The setting for high high alarm in engineering units.
37	HI_PRI	Unsigned8	0 to 15	0		S	Priority of the high alarm.
38	HI_LIM	Float	OUT_SCALE, +INF	+INF	PV	S	The setting for high alarm in engineering units.
39	HI_LIMX	Float	OUT_SCALE, +INF	+INF	PV	S	The setting for high alarm in engineering units.
40	LO_PRI	Unsigned8	0 to 15	0		S	Priority of the low alarm.
41	LO_LIM	Float	OUT_SCALE, -INF	-INF	PV	S	The setting for low alarm in engineering units.
42	LO_LIMX	Float	OUT_SCALE, -INF	-INF	PV	S	The setting for low alarm in engineering units.
43	LO_LO_PRI	Unsigned8	0 to 15	0		S	Priority of the low low alarm.
44	LO_LO_LIM	Float	OUT_SCALE, -INF	-INF	PV	S	The setting for low low alarm in engineering units.
45	LO_LO_LIMX	Float	OUT_SCALE, -INF	-INF	PV	S	The setting for low low alarm in engineering units.
46	HI_HI_ALM	DS-71			PV	D	The status for high high alarm and its associated time stamp.

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
47	HI_ALM	DS-71			PV	D	The status for high alarm and its associated time stamp.
48	LO_ALM	DS-71			PV	D	The status for low alarm and its associated time stamp.
49	LO_LO_ALM	DS-71			PV	D	The status for low low alarm and its associated time stamp.

Legend: *E – Enumerated parameter; na – Adimensional parameter; RO – Read only; D – dynamic; N – non-volatile; S - static*

If BEHAVIOR parameter is “Adapted”:

The required mode for writing is the actual mode, regardless the target mode : OUT

ISEL - Input Selector

Description

The signal selector block provides selection of up to four inputs and generates an output based on the configured action. This block would normally receive its input from an AI or another block and not directly from a transducer. In addition to signal selection the block can also perform max, min, mid, avg and first good selection. With the combination of parameter configuration options the block can function as a rotary position switch, or a validated priority selection based on the use of the first good parameter and the DISABLE_n parameter. As a switch the block can receive switch toggle information from both the connected inputs or from an operator input. The block also supports the concept of a middle selection. Logic is provided for handling of dubious and bad signals in conjunction with configured actions. The intended application of this block is to provide control signal selection in the forward path only and is not intended to receive signals from the output of a controller, therefore, no back calculation support is provided.

The algorithm has the following actions, choose by the SELECT_TYPE parameter:

Max = select the max from all the connected and good inputs

Min = select the min from all the connected and good inputs

Mid = select the middle value from all the connected and good inputs, if less than MIN_GOOD inputs are connected then an error code is generated. An error code is also generated if less than MIN_GOOD values have good status. Not intended for use with 2 or 4 inputs. Although the normal configuration for this feature would be with three signals the block will generate an average of the middle two if four signals are configured or the average of two if three are configured and a bad status is passed on one of the inputs.

First Good = determine the first good input encountered based on ascending evaluation of the inputs, see text for further discussion

Avg = compute the average for all the connected and good inputs, if less than two inputs are connected then set the output equal to the input and generate an error code. An error code is also generated if less than MIN_GOOD inputs have good status.

The processing of the block is as follows:

If DISABLE_n is true then the respective input IN_n is not used.

Inputs whose status is bad are ignored. Process the dubious option.

MIN_GOOD specifies the minimum available signals for OUT status to be good. If there are no inputs left, or fewer than MIN_GOOD inputs then set the status of OUT to Bad and the value of SELECTED to zero. Do not do selection processing.

For Average ("Avg") the status of OUT is set to worst of inputs considered, then average of the useable inputs is calculated, if none are useable then the output status is bad.

When SELECT_TYPE is "Mid", "First Good", "Max", or "Min", set the status of OUT to the status of the selected input. The "First Good" option starts at the first input then the second until it encounters an input whose DISABLE_n is not set and its status is Good, and then transfers this value to the output of the block.

If OP_SELECT is non-zero, the OP_SELECT value will override the SELECT_TYPE selection.

SELECTED is a second output which will indicate which input has been selected by the algorithm for all SELECT_TYPE except "Avg", when it reflects the number of inputs used in the average. The SELECTED output always has a good status, unless the block is out of service.

BLOCK_ERR

The BLOCK_ERR of the ISEL block will reflect the following causes:

Block Configuration Error – the configuration error occurs when the SELECT_TYPE parameter has an invalid value;

Out of Service – it occurs when the block is in O/S mode.

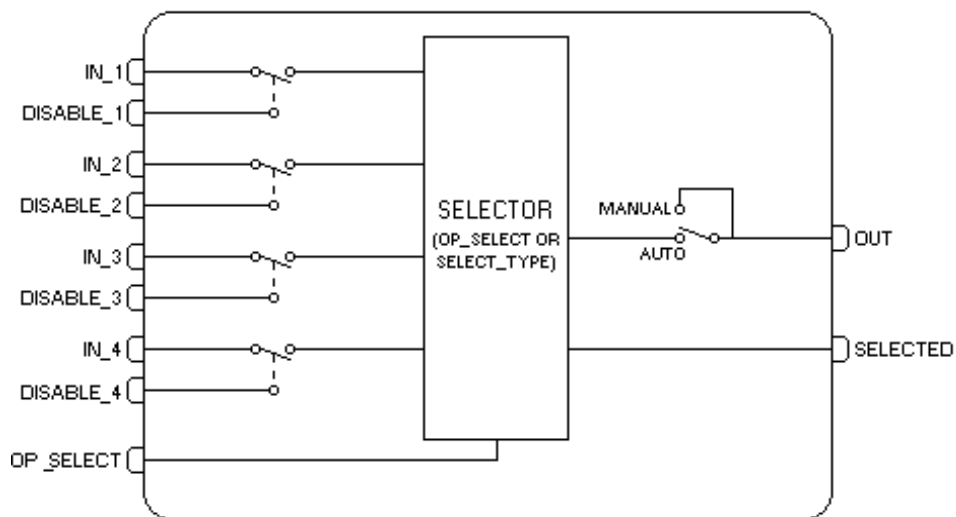
Supported Modes

O/S, MAN and AUTO.

Status Handling

The status of OUT will be a copy of the selected input, but if the output is an average of inputs the status will be Good Non-cascade – Non-specific.

Schematic



Parameters

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
4	ST_REV	Unsigned16		0	None	S	
5	TAG_DESC	OctString(32)		Spaces	Na	S	
6	STRATEGY	Unsigned16		0	None	S	
4	ALERT_KEY	Unsigned8	1 to 255	0	None	S	
5	MODE_BLK	DS-69		O/S	Na	S	See Mode Parameter
6	BLOCK_ERR	Bitstring(2)			E	D	
7	OUT	DS-65	XD_SCALE		OUT	D / Man	The primary analog value calculated as a result of executing the function.
8	OUT_UNITS	Unsigned16		0	E	S	The engineering units of the output for display.
9	GRANT_DENY	DS-70		0	Na	D	Options for controlling access of host computer and local control panels to operating, tuning and alarm parameters of the block.

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
10	STATUS_OPTS	Bitstring(2)	See Block Options	0	Na	S / O/S	See Block Options
11	IN_1	DS-65				D	Numbered input parameter 1.
12	IN_2	DS-65				D	Numbered input parameter 2.
13	IN_3	DS-65				D	Numbered input parameter 3.
14	IN_4	DS-65				D	Numbered input parameter 4.
15	DISABLE_1	DS-66				D	Parameter to switch off the input from being used. If this parameter is true then don't use this input IN_1 determining the output.
16	DISABLE_2	DS-66				D	Parameter to switch off the input from being used. If this parameter is true then don't use this input IN_2 determining the output.
17	DISABLE_3	DS-66				D	Parameter to switch off the input from being used. If this parameter is true then don't use this input IN_3 determining the output.
18	DISABLE_4	DS-66				D	Parameter to switch off the input from being used. If this parameter is true then don't use this input IN_4 determining the output.
19	SELECT_TYPE	Unsigned8	1=First good 2=MIN 3=MAX 4=MID 5=AVG	0	E	S	selector action max = select the max from all the connected and good inputs min = select the min from all the connected and good inputs mid = select the mid value from all the connected and good inputs, if less than min_good inputs are connected then an error code is generated. An error code is also generated if less than min_good values have good status. Not intended for use with 2 or 4 inputs. First Good = determine the first good input encountered based on ascending evaluation of the inputs, see text for further discussion avg = compute the average for all the connected and good inputs, if less than two inputs are connected then set the output equal to the input and generate an error code. An error code is also generated if less than min_good inputs have good status
20	MIN_GOOD	Unsigned8	0 through 4	0		S	If the number of inputs which are good is less than the value of MIN_GOOD then set the out status to bad.

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
21	SELECTED	DS-66	None, 1= 1 2= 2 3= 3 4= 4			D / RO	An integer indicating which input has been selected.
22	OP_SELECT	DS-66	0 thru 4		None	D	An operator adjustable parameter to force a given input to be used.
23	UPDATE_EVT	DS-73			na	D	This alert is generated by any change to the static data.
24	BLOCK_ALM	DS-72			na	D	The block alarm is used for all configuration, hardware, connection failure or system problems in the block. The cause of the alert is entered in the subcode field. The first alert to become active will set the Active status in the Status attribute. As soon as the Unreported status is cleared by the alert reporting task, another block alert may be reported without clearing the Active status, if the subcode has changed.

Legend: *E* – Enumerated parameter; *na* – Adimensional parameter; *RO* – Read only; *D* – dynamic;
N – non-volatile; *S* - static

If BEHAVIOR parameter is “Adapted”:

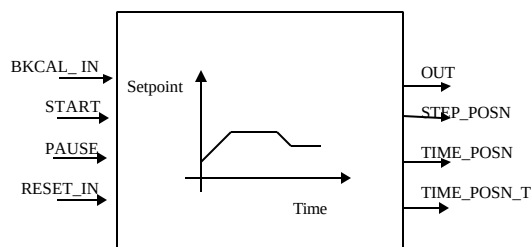
The default value of SELECT_TYPE is First good.

The required mode for writing is the actual mode, regardless the target mode : OUT

SPG - Setpoint Ramp Generator

Description

The Setpoint generator block is normally used to generate a Setpoint to a PID block in applications like temperature control, batch reactors, etc. In those applications, the Setpoint shall follow a certain profile in function of the time.



The block algorithm shall comply with the following:

1) The profile is determined by up to ten segments or steps. Each segment is defined by a starting value [START_VAL] and a time duration [DURATION]. The starting value of the next segment determines if the previous segment ramps up, down or remains constant. The profile is given by two parallel arrays and a parameter for the time unit:

START_VAL (Starting value) - Eleven floating point values defining the initial value of each step, in engineering units.

DURATION (Time duration) - Ten floating point values defining the duration, in seconds, of each step. A null value defines the last step.

TIME_UNITS - A contained unsigned-eight parameter used to specify the time units used for display.

2) The two arrays define the Setpoint value (y-axis) in function of the time (t-axis). Between two given points, the Setpoint is calculated by interpolation. As each segment is defined by [START_VAL]_i, [DURATION]_i and [START_VAL]_{i+1}, a profile with 'n' segments will need **n+1** starting values and n time durations. As example, the two following arrays define the profile shown on Fig. 1:

	1	2	3	4	5	6
START_VAL	25	50	50	100	100	25
DURATION	60	60	120	60	60	0

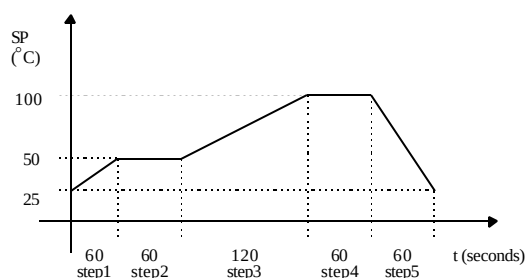


Fig.1 - Setpoint profile

- 3) The timer is started by a transition from false to true at input START.
- 4) The timer may be interrupted at any time by changing the discrete signal PAUSE from false to true. It will resume running when PAUSE is set to false. The PAUSE will not force *manual* mode.
- 5) The timer is also interrupted by a PAUSE caused by the deviation between BKCAL_IN and the generated Setpoint. If the deviation exceeds DV_HI_LIM or DV_LO_LIM, an alarm is indicated in DV_HI_ALM or DV_LO_ALM, respectively. Both alarms stop the timer and resume normal operation when the deviation is within the prescribed limits.
- 6) The Setpoint is in the “y” axis, while the time is in the “t” axis. The Setpoint value is available at output OUT. It is also available in PRE_OUT even when the block is in Man mode. For display purpose, the engineering unit of OUT is given by OUT_SCALE.
- 7) Three outputs inform the current point of the profile:

STEP_POSN - Informs the current segment or step.
TIME_POSN - Informs the time elapsed since the beginning of the current step.
TIME_POSN_T - Informs the time elapsed since the beginning of the profile.
- 8) With the block in manual, the operator can write on the outputs STEP_POSN, TIME_POSN and TIME_POSN_T in order to select a particular point of the profile. When the block is switched back to auto, the profile will start from that point. The timer is restarted by activating the input START.
- 9) With the block in manual, the operator can also modify OUT. As the adjusted value may correspond to more than one point on the profile or to none, if the operator adjusts a value beyond the profile limits, the OUT value goes from the last adjusted value to the point before mode switching following a ramp defined by BAL_TIME.
- 10) Another operation that can be done with the block in manual, is to advance or return the time through the following operator commands (OP_CMD_SPG):

ADVANCE - sets the time to the beginning of the next step.
REPEAT - sets the time to the beginning of the current step.
- 11) The outputs can only be modified with the block is in manual mode.
- 12) The operator can give a RESET command using OP_CMD_SPG with the block in any mode. The timer is set to zero, i.e., to the beginning of the profile. In this case the operator must give a new start, by switching the input START from false to true. Therefore the block may be started even though the OP_CMD_SPG remains with RESET value.
- 13) The input RESET_IN allows a discrete signal coming from another block to set the timer to zero. While this input parameter has value TRUE, the block will remain in reset, therefore it will be able to start only after this input parameter goes to FALSE.
- 14) When the time reaches the last point of the profile, it will automatically return to zero (RESET) and restart (START) automatically, if the parameter AUTO_CYCLE is set to true.
- 15) The operation status is given by the parameter SPG_STATE

READY - When the profile is at the beginning, waiting for the starting signal.
ACTIVE - When the timer is “on”.
PAUSE - When the PAUSE signal stopped the timer.
AT_END - When the time reaches the last point of the profile.
- 16) The parameter PAUSE_CAUSE enumerates the cause of the PAUSE state:

- 1 = Operator Pause
- 2 = Logic Pause
- 3 = Operator & Logic
- 4 = Deviation pause
- 5 = Operator & Deviation
- 6 = Logic & Deviation
- 7 = Operator & Logic & Deviation

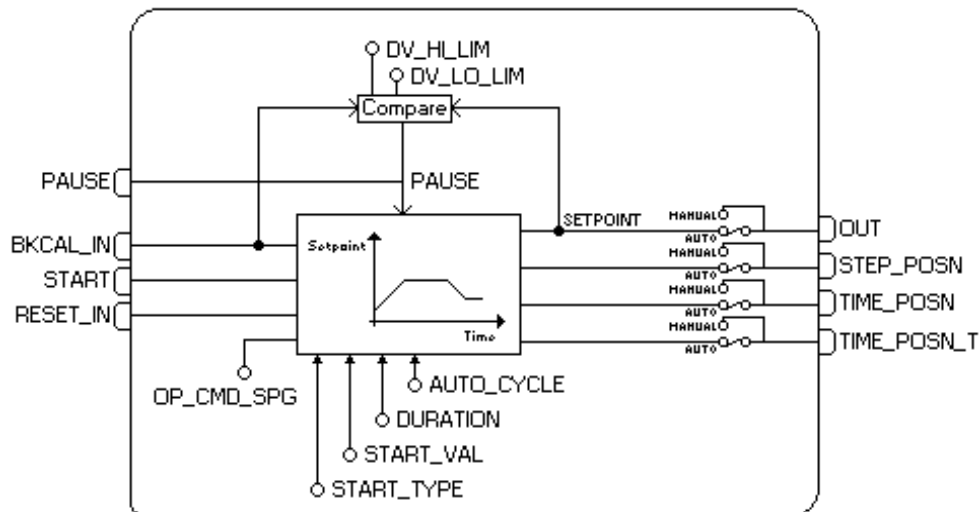
Logic Pause happens when the deviation limits are exceeded or the PID block is not in Cascade mode.

17) Sometimes there is a large deviation between the controlled variable (available in BKCAL_IN) and the profile initial value. In this situation, the timer may not be started or the control will start with a large upset. In order to avoid these problems, the parameter START_TYPE offers the following options:

- a - USE_CURVE- The curve starts as specified by START_VAL and DURATION.
- b - USE_DUR- The curve starts at BKCAL_IN value and use the duration specified.
- c - USE_RATE- The curve starts at BKCAL_IN value and use the rate specified by the first two START_VAL values and the first DURATION value.

18) The input BKCAL_IN can be connected to the output of an Analog Input block or to the BKCAL_OUT of a PID control block. If a PID is connected, the CONTROL_OPTS of the PID should be configured to use PV for BKCAL_IN. If the PID is not in Cas mode when the operation status is READY (see 15), initialization will occur as described in 17. If the operation status is ACTIVE, the block will go to IMan mode and behave as described in 9 to make the value of OUT equal the value of BKCAL_IN.

SCHEMATIC



BLOCK_ERR

The BLOCK_ERR of the SPG block will reflect the following causes:
 Block Configuration Error – the configuration error occurs when the START_TYPE parameter has an invalid value;
 Out of Service – it occurs when the block is in O/S mode.

Supported Modes

O/S, IMAN, MAN and AUTO.

Status Handling

The input BKCAL_IN provides status.

If the status of BKCAL_IN is bad and the option Use uncertain of STATUS_OPTS is set to true, the deviation alarms are not considered.

If the status of any used input becomes bad or uncertain and the respective option Use bad or Use uncertain of STATUS_OPTS is not set, the block actual mode will be forced to manual.

Parameters

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
1	ST_REV	Unsigned16		0	None	S	
2	TAG_DESC	OctString(32)		Spaces	Na	S	
3	STRATEGY	Unsigned16		0	None	S	
4	ALERT_KEY	Unsigned8	1 to 255	0	None	S	
5	MODE_BLK	DS-69		O/S	Na	S	See Mode Parameter
6	BLOCK_ERR	Bitstring(2)			E	D / RO	
7	OUT	DS-65	OUT_SCALE +/- 10%		OUT	N / Man	The analog value calculated as a result of executing the function.
8	OUT_SCALE	DS-68		0-100%	OUT	S / Man	The high and low scale values to the OUT parameter.
9	GRANT_DENY	DS-70		0	na	D	Options for controlling access of host computer and local control panels to operating, tuning and alarm parameters of the block.
10	STATUS_OPTS	Bitstring(2)	See Block Options	0	Na	S / O/S	See Block Options
11	START_VAL	11 Floats					An array of up to eleven points defines the starting point of each segment of the Setpoint profile, in engineering units.
12	DURATION	10 Floats		0's	Sec	S	An array of up to ten points defines the duration of each segment of the Setpoint profile, in seconds.
13	TIME_UNITS	Unsigned8	1=seconds 2=minutes 3=hours 4=days 5=[day- [hr:[min[:sec]]]]	0		E	Display Time Units for TIME_POSN and TIME_POSN_T.
14	BKCAL_IN	DS-65			OUT	N	The value and status from a lower block's BKCAL_OUT that is used to prevent reset windup and to initialize the control loop.

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
15	START	DS-66			On/Off	D	A leading edge at this input or a transition from false to true starts the timer.
16	START_TYPE	Unsigned8	1=Use Curve 2=Use Duration 3=Use Rate	0	E	S	This parameter selects the starting point option.
17	PAUSE	DS-66				D	Stops the timer when set to <i>true</i> . Resume time running when set back to <i>false</i> .
18	PAUSE_CAUSE	Unsigned8	0=Not paused 1=Operator Pause 2=Logic Pause 4=Deviation Pause 3=Operator & Logic 5=Operator & Deviation 6=Logic & Deviation 7=Operator & Logic & Deviation			E	This parameter enumerates the causes of PAUSE.
19	AUTO_CYCLE	Unsigned8	1:Auto cycle	0	E	S	When set to <i>true</i> , automatically resets the time to the beginning of the first step and restarts the timer.
20	STEP_POSN	DS-66	0=none 1=step1 2=step 2 n=step n	0	E	D / Man	Determine the current step or segment of the profile in auto mode. Direct the timer to the step specified by the operator when in manual mode.
21	TIME_POSN	DS-65			Sec	D / Man	Determine the time elapsed since the beginning of the step in auto mode. The operator can set the time since the beginning of the current step when operating in manual.
22	TIME_POSN_T	DS-65			Sec	N / Man	Determine the time elapsed since the beginning of the curve in auto mode. The operator can set the time since the beginning of the curve when operating in manual.
23	OP_CMD_SPG	Unsigned8	0=UNDEFINED 1=RESET_IN 2=ADVANCE 3=REPEAT	0	E	D	Enable the positioning in the profile. Enumerations are: RESET, ADVANCE*, REPEAT*. (*only valid with the block in manual).

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
24	SPG_STATE	Unsigned8	0=UNDEFINED 1=READY 2=ACTIVE 3=PAUSE 4=AT_END		E	N	Define the operating state of the block. Enumerations are: READY, ACTIVE, PAUSE and AT_END.
25	PRE_OUT	DS-65				D	Displays what would be the OUT value and status if the mode was Auto or lower.
26	RESET_IN	DS-66	0:Off 1:Reset		E	D	Resets the timer.
27	BAL_TIME	Float	Positive	0	sec	S	This specifies the time for the internal working value of bias or ratio to return to the operator set bias or ratio, in seconds.
28	OUTAGE_LIM	Float	Positive	0	Sec	S	The maximum tolerated duration for power failure. This feature is not supported.
29	UPDATE_EVT	DS-73			Na	D	This alert is generated by any change to the static data.
30	BLOCK_ALM	DS-72			Na	D	The block alarm is used for all configuration, hardware, connection failure or system problems in the block. The cause of the alert is entered in the subcode field. The first alert to become active will set the Active status in the Status attribute. As soon as the Unreported status is cleared by the alert reporting task, another block alert may be reported without clearing the Active status, if the subcode has changed.
31	ALARM_SUM	DS-74	See Block Options		Na	S	The current alert status, unacknowledged states, unreported states, and disabled states of the alarms associated with the function block.
32	ACK_OPTION	Bitstring(2)	0: Auto ACK Disable 1: Auto ACK Enable	0	Na	S	Selection of whether alarms associated with the block will be automatically acknowledged
33	ALARM_HYS	Float	0 to 50 %	0.5%	%	S	Alarm hysteresis parameter. In order to clear the alarm the amount the PV must return within the alarm limit plus hysteresis.
34	DV_HI_PRI	Unsigned8	0 to 15			S	Priority of the high deviation alarm.
35	DV_HI_LIM	Float	OUT_SCALE, +INF	+INF	OUT	S	The setting for high deviation alarm in engineering units.
36	DV_LO_PRI	Unsigned8	0 to 15			S	Priority of the low deviation alarm.
37	DV_LO_LIM	Float	OUT_SCALE, -INF	-INF	OUT	S	The setting for low deviation alarm in engineering units.

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
38	DV_HI_ALM	DS-71			OUT	D	The status for high deviation alarm and its associated time stamp.
39	DV_LO_ALM	DS-71			OUT	D	The status for low deviation alarm and its associated time stamp.

Legend: *E* – Enumerated parameter; *na* – Adimensional parameter; *RO* – Read only; *D* – dynamic;
N – non-volatile; *S* – static

If BEHAVIOR parameter is “Adapted”:

The default value of START_TYPE is “Use Curve”.

The required mode for writing is the actual mode, regardless the target mode : OUT, TIME_POSN, TIME_POSN_T and STEP_POSN

TIME – Timer and Logic

Description

The Timer and Logic function block provides logic combination and timing functions including the following:

- Combine multiple inputs as OR, AND, vote, or EXACTLY count.
- Measure the duration of the combined discrete input signal
- Accumulate, until reset, the duration of the combined input signal
- Count changes of the combined discrete input signal
- Set a discrete output if the duration of the combined input signal exceeds a limit
- Extend, Delay, Pulse, or Debounce the combined input as an output
- Provide outputs indicating amount of time expired and amount of time remaining
- Selectively invert any connected discrete input or output
- Reset timer

Up to four inputs may be combined logically (ANDed, ORed), voted (any 2 or more true, any 3 or more true), or counted (exactly 1 true, exactly 2 true, exactly 3, odd count, or even count). The combined input value is specified by the combination type (COMB_TYPE) enumeration. Choices are indicated in the table below.

Connected inputs may have the values of true, false, or undefined. Undefined connected inputs are treated with status bad(out-of-service). Non-connected inputs may have the values of true, false, or undefined. Undefined non-connected inputs (operator/engineer enterable) are ignored.

COMB_TYPE	PV_D value
Enumeration	
OR	true if one or more used inputs are true
ANY2	true if two or more used inputs are true
ANY3	true if three or more used inputs are true
AND	true if all used inputs are true
EXACTLY1	true if exactly 1 used input is true
EXACTLY2	true if exactly 2 used inputs are true
EXACTLY3	true if exactly 3 used inputs are true
EVEN	true if exactly 0, 2 or 4 used inputs are true (Exclusive-OR)
ODD	true if exactly 1 or 3 used inputs are true (Not-Exclusive-OR)

The timer processing type is specified by `TIMER_TYPE`. It may operate to produce a measurement, delay, extension, pulse (non-re-triggerable or re-triggerable) or debounce, of the combined input signal.

`TIMER_SP` is the specification for the time duration of delay, extension, pulse, debounce filter, or comparison limit. It may either be configured as an operator/engineer-entered constant or may be connected as an input, determined by another block. In either case the block will, on each execution, check to see if the current duration of the delay, extension, pulse, debounce, or time comparison exceeds the current `TIMER_SP`.

`OUT_EXP` indicates the amount of time expired in the measurement, comparison, delay, extension, debounce, or pulse. See `TIMER_TYPE` for details.

`OUT_REM` indicates the amount of time remaining in the comparison, delay, extension, debounce, or pulse. See `TIMER_TYPE` for details.

`QUIES_OPT` allows the configurer to select the behavior for `OUT_EXP` and `OUT_REM` when the timer is quiescent-- that is, not timing and not in a triggered condition. The following table lists the definition of quiescent state for each `TIMER_TYPE` enumeration:

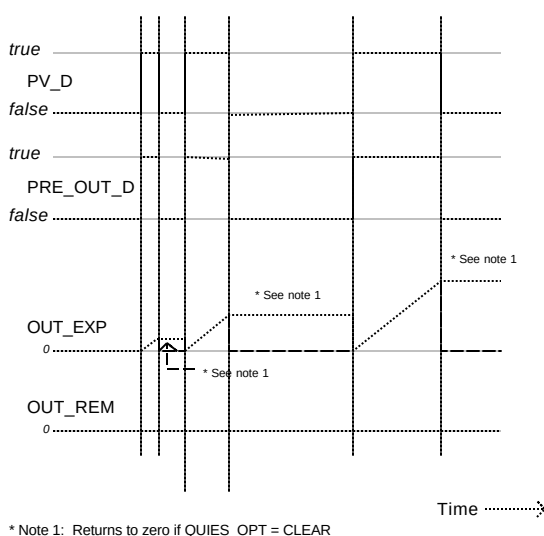
Definition of quiescent state start and end as a function of <code>TIMER_TYPE</code>		
<code>TIMER_TYPE</code>	Quiescence state starts when combined input (<code>PV_D</code>):	Quiescence state ends when combined input (<code>PV_D</code>):
MEASURE	returns to false	changes from false-to-true
ACCUM	[<code>QUIES_OPT</code> does not apply]	[<code>QUIES_OPT</code> does not apply]
COMPARE	returns to false	changes from false-to-true
DELAY	returns to false	changes from false-to-true
EXTEND	returns to true	changes from true-to-false
DEBOUNCE	has changed <u>and</u> timer has expired	changes
PULSE	has returned to false <u>and</u> timer has expired	changes from false-to-true
RT_PULSE	has returned to false <u>and</u> timer has expired	changes from false-to-true

The `CLEAR` enumeration of `QUIES_OPT` will cause both `OUT_EXP` and `OUT_REM` to be set to zero during quiescence. The `LAST` enumeration of `QUIES_OPT` will cause both `OUT_EXP` and `OUT_REM` to be held to their values when the block becomes quiescent. That is, the time expired and time remaining will remain available until the quiescence ends with the start of the next activation. Note that a false-to-true transition on `RESET_IN` will also reset `OUT_EXP` and `OUT_REM`.

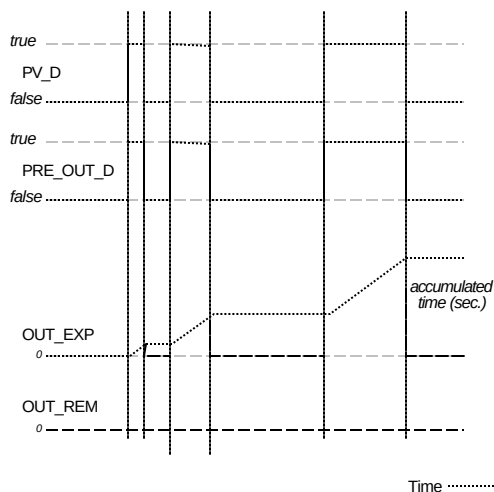
`N_START` is a count of the number of starts (false-to-true) transitions of the combined input, `PV_D`, since the last false-to-true change seen on `RESET_IN`.

TIMER_TYPE may be one of the following, operating on the combined input signal:

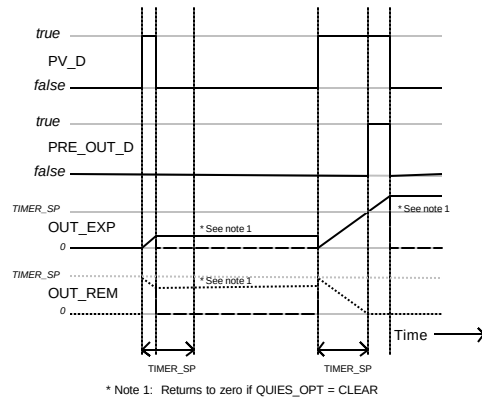
- **MEASURE** Indicate the duration of the most recent true signal
 - **ACCUM** Accumulate the durations of a true signal
 - **COMPARE** Compare a true signal duration to specified duration
 - **DELAY** Delay a false-to-true transition, eliminating it if short
 - **EXTEND** Extend a true-to-false transition, eliminating it if short
 - **DEBOUNCE** Delay any transition, eliminating it if short
 - **PULSE** Generate a true pulse on a false-to-true transition, non-retriggerable
 - **RT_PULSE** Generate a true pulse on a false-to-true transition, retriggerable
- If **TIMER_TYPE** is **MEASURE**, **PRE_OUT_D** will be the same as the combined input, **PV_D**. **OUT_EXP** indicates the length of time, in seconds, that the combined signal is true. **OUT_REM** is set to 0.



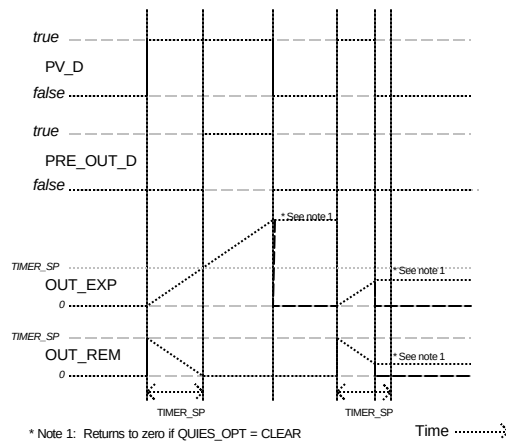
- If **TIMER_TYPE** is **ACCUM**, **PRE_OUT_D** will be the same as the combined input, **PV_D**. **OUT_EXP** indicates the accumulated length of time, in seconds, that the combined signal has been true. Unlike **TIMER_TYPE** = MEAS, it will not be automatically reset by the time of the next occurrence of a false-to-true change of **PV_D**. Instead, it will continue to accumulate "on" time or "run" time until reset to 0 by a false-to-true change on **RESET_IN**. **OUT_REM** is unused (set to 0.0) for this timer type.



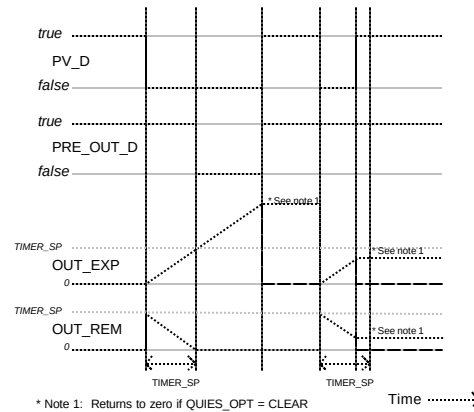
- If **TIMER_TYPE** is **COMPARE**, the block will measure the time since a false-to-true change on the combined input, **PV_D**. The current duration will be indicated by **OUT_EXP**. **OUT_REM** will indicate the time remaining between the current expired duration, **OUT_EXP**, and current limit, **TIMER_SP**. If **OUT_EXP** does not exceed **TIMER_SP**, **PRE_OUT_D** will be set to false. If **OUT_EXP** equals or exceeds **TIMER_SP**, **PRE_OUT_D** will be set to true and **OUT_REM** will be set to zero. When the combined input returns to false, either with or without exceeding the limits specified by **TIMER_SP**, **OUT_D** will be set to false. [Note that this type of behavior is the same as **TIMER_TYPE** = **DELAY**. The difference is merely in the application perspective.]



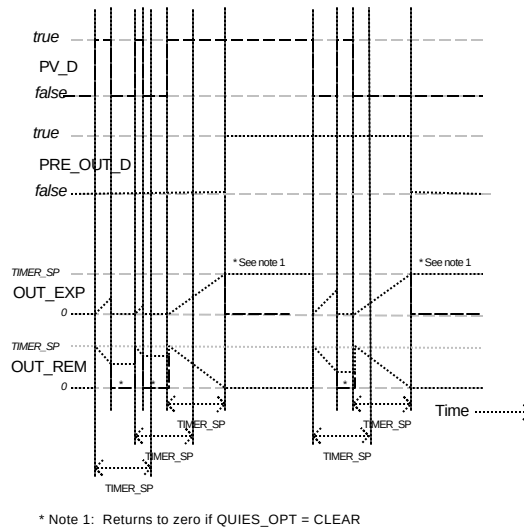
- If **TIMER_TYPE** is **DELAY**, a false-to-true change on the combined input, **PV_D**, will be delayed at the output, **PRE_OUT_D**, until the amount of time specified by **TIMER_SP** has been expired. If the combined input returns to false before the time expires, the output will remain as false, concealing the input transitions. If the **PRE_OUT_D** output has been set to true because the time has expired, a true-to-false transition in the combined input will be presented to **PRE_OUT_D** immediately. [Note that this type of behavior is the same as **TIMER_TYPE** = **COMPARE**. The difference is merely in the application perspective.]



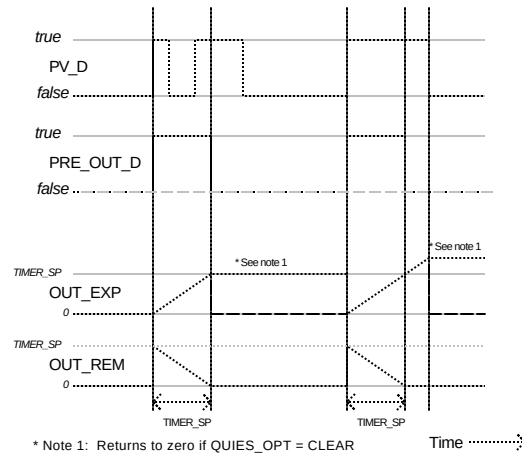
- If **TIMER_TYPE** is **EXTEND**, a true-to-false change on the combined input, **PV_D**, will be delayed at the output, **PRE_OUT_D**, until the amount of time specified by **TIMER_SP** has been expired. If the combined input returns to true before the time expires, the output will remain as true, concealing the input transitions. If the **PRE_OUT_D** output has been set to false because the time has expired, a false-to-true transition in the combined input will be presented to **PRE_OUT_D** immediately.



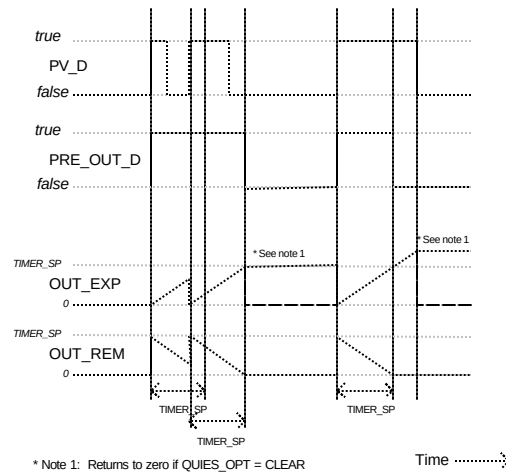
- If **TIMER_TYPE** is **DEBOUNCE**, and if **PRE_OUT_D** is false, a false-to-true change on the combined input, **PV_D**, will be delayed at the output, **PRE_OUT_D**, until the amount of time specified by **TIMER_SP** has been expired. If the combined input returns to false before the time expires, the output will remain as false, concealing the input transitions. If **PRE_OUT_D** is true, a true-to-false change on the combined input, **PV_D**, will be delayed at the output, **PRE_OUT_D**, until the amount of time specified by **TIMER_SP** has been expired. If the combined input returns to true before the time expires, the output will remain as true, concealing the input transitions. This both delays true initiations and extends true terminations, acting as a filter for intermittent state changes.



- If **TIMER_TYPE** is **PULSE**, a false-to-true change on the combined input, **PV_D**, will initiate a true pulse at **PRE_OUT_D** whose duration is determined by the **TIMER_SP** value. At the end of the time duration, the output, will return to false. Further false-to-true transitions of the combined input while **PRE_OUT_D** is true will be ignored.



- If **TIMER_TYPE** is **RT_PULSE**, (Re-Triggerable pulse type) a false-to-true change on the combined input, **PV_D**, will initiate a true pulse at **PRE_OUT_D** whose duration is determined by the **TIMER_SP** value. At the end of that time duration **PRE_OUT_D** will return to false. If the combined input returns to false and presents a subsequent false-to-true transition while the timer is timing, the timer shall be reinitialized and **PRE_OUT_D** shall continue to be true.



RESET_IN is a discrete input which, on a false-to-true transition, resets the timer. **OUT_EXP** is set to 0.0, and then the timer follows processing described under "Initial Value Handling" regarding the value of **PRE_OUT_D** and **OUT_REM**. If **RESET_IN** is not connected, an operator/engineer may set it to true. In this case, the block logic will reset it to false on its next execution.

TIME_UNITS allows the user to specify to the HMI the units of time in which **TIMER_SP**, **OUT_EXP** and **OUT_REM** are to be displayed.

Each bit in **INVERT_OPTS**, if set, indicates that the corresponding discrete-with-status input or output parameter is inverted. That is, input values are inverted prior to use by the block and outputs are inverted after the value is determined by the block.

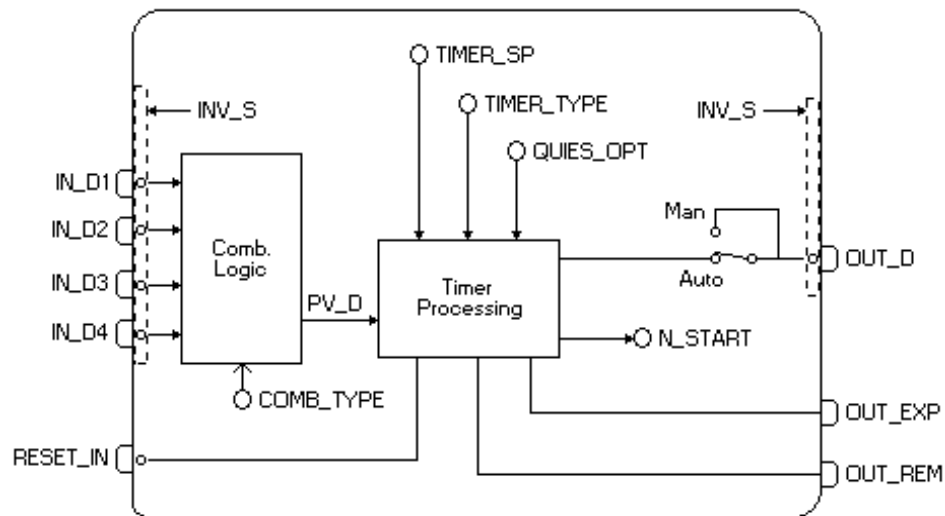
BLOCK_ERR

The BLOCK_ERR of the TIME block will reflect the following causes:
 Block Configuration Error – the configuration error occurs when the TIME_UNITS or QUIES_OPT parameters have an invalid value;
 Out of Service – it occurs when the block is in O/S mode.

Supported Modes

O/S, MAN and AUTO.

Schematic



Parameters

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
1	ST_REV	Unsigned16		0	None	S	
2	TAG_DESC	OctString(32)		Spaces	Na	S	
3	STRATEGY	Unsigned16		0	None	S	
4	ALERT_KEY	Unsigned8	1 to 255	0	None	S	
5	MODE_BLK	DS-69		O/S	Na	S	See Mode Parameter
6	BLOCK_ERR	Bitstring(2)			E	D	This is the timer duration used by the timer block for delay, extension, debouncing, and pulse time-processing.
7	PV_D	DS-66				D	Either the primary discrete value for use in executing the function, or a process value associated with it.

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
8	OUT_D	DS-66				D	The primary discrete value calculated as a result of executing the function.
9	TIMER_SP	Float	Positive	0	Sec	S	
10	PV_STATE	Unsigned16		0		S	Index to the text describing the states of a discrete PV.
11	OUT_STATE	Unsigned16		0		S	Index to the text describing the states of a discrete output.
12	GRANT_DENY	DS-70		0	Na	D	Options for controlling access of host computer and local control panels to operating, tuning and alarm parameters of the block.
13	INVERT_OPTS	Bitstring(2)	See Block Options	0	Na	S / O/S	See Block Options
14	STATUS_OPTS	Bitstring(2)	See Block Options	0	Na	S / O/S	See Block Options
15	IN_D1	DS-66				D	Numbered discrete input parameter 1.
16	IN_D2	DS-66				D	Numbered discrete input parameter 2.
17	IN_D3	DS-66				D	Numbered discrete input parameter 3.
18	IN_D4	DS-66				D	Numbered discrete input parameter 4.
19	COMB_TYPE	Unsigned8	0=AND 1=OR 2=ANY2 3=ANY3 21=EXACTLY1 22=EXACTLY2 23=EXACTLY3 40=EVEN 41=ODD	1	E	S	Determines how the multiple IN_D[] values are combined.
20	TIMER_TYPE	Unsigned8	0=MEASURE 1=ACCUM 2=COMPARE 3=DELAY 4=EXTEND 5=DEBOUNCE 6=PULSE 7=RT_PULSE	0	E	S	Type of time-processing applied to PV_D to determine the PRE_OUT_D.
21	PRE_OUT_D	DS-66				D	This parameter is the combined and time-processed output of the timer block.
22	N_START	Unsigned16			None	D	Count of <i>false-to-true</i> transitions of the combined input, PV_D. Reset by <i>false-to-true</i> transition of RESET_IN.

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
23	OUT_EXP	DS-65			Sec	N / RO	This is the time expired. Stops when TIMER_SP is reached. Reset to zero (1) by RESET_IN, (2) at start of next timer event if QUIES_OPT = LAST, or (3) when block becomes quiescent if QUIES_OPT = CLEAR.
24	OUT_REM	DS-65			Sec	N / RO	This is the time remaining if the timer is active. Stops when event ceases (block becomes quiescent). Reset to 0.0 if QUIES_OPT = CLEAR, and the timer is inactive.
25	RESET_IN	DS-66	0=Off 1=Reset				Resets the timer.
26	QUIES_OPT	Unsigned8	1=CLEAR 2=LAST	0	E	S / O/S	Behavior option for OUT_EXP and OUT_REM during quiescence. CLEAR resets them to zero. LAST causes last values to be held.
27	TIME_UNITS	Unsigned8	1=seconds 2=minutes 3=hours 4=days 5=[day- [hr:[min[:sec]]]]	0	E	S	Display Time Units for TIMER_SP, OUT_EXP, and OUT_REM:
28	UPDATE_EVT	DS-73			na	D	This alert is generated by any change to the static data.
29	BLOCK_ALM	DS-72			na	D	The block alarm is used for all configuration, hardware, connection failure or system problems in the block. The cause of the alert is entered in the subcode field. The first alert to become active will set the Active status in the Status attribute. As soon as the Unreported status is cleared by the alert reporting task, another block alert may be reported without clearing the Active status, if the subcode has changed.

Legend: E – Enumerated parameter; na – Adimensional parameter; RO – Read only; D – dynamic;
N – non-volatile; S - static

If BEHAVIOR parameter is “Adapted”:

The default value of TIME_UNITS is “Seconds”.
The default value of QUIES_OPT is “CLEAR”.

LLAG - Lead Lag

Description

The LLAG block provides dynamic compensation of the IN parameter. The block can function as a lead or lag device. The user would configure the LEAD_TIME and LAG_TIME parameters to obtain the desired input/output relationship. This block would normally be used in a feedforward portion of a control scheme or used to implement some special initialization functions required by a control scheme. This block will normally participate in a feedforward path. The FOLLOW parameter is used to cause the block to perform tracking functions whereby the output is forced to track the input whenever the FOLLOW parameter is set true.

The LAG_TIME parameter specifies the time constant for the block. Based on a step change to the input this is the time to reach 63.2% of the final value. Normally, it requires five time constants to reach the final value based on a first order function applied to the input. The LEAD_TIME parameter specifies the gain or impulse applied to the input parameter. The generalized form of the equation describing the action is as follows:

$$(T_1s+1)/(T_2s+1)$$

where:

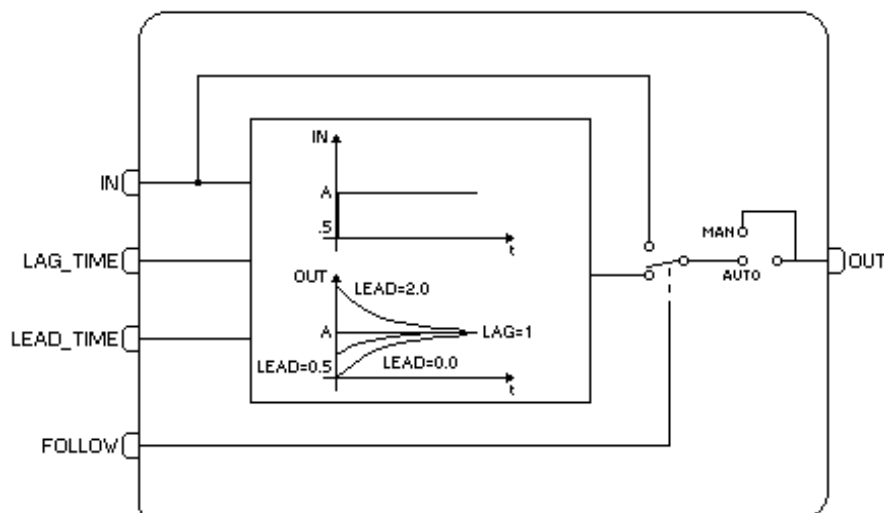
T_1 = Lead time Constant

T_2 = Lag Time Constant

Supported Modes

O/S, MAN and AUTO.

Schematic



Parameters

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
1	ST_REV	Unsigned16		0	None	S	
2	TAG_DESC	OctString(32)		Spaces	Na	S	
3	STRATEGY	Unsigned16		0	None	S	
4	ALERT_KEY	Unsigned8	1 to 255	0	None	S	
5	MODE_BLK	DS-69		O/S	Na	S	See Mode Parameter
6	BLOCK_ERR	Bitstring(2)			E	D	
7	OUT	DS-65			OUT	D	The primary analog value calculated as a result of executing the function.
8	OUT_UNITS	Unsigned16		0	E	S	The maximum tolerated duration for power failure.
9	GRANT_DENY	DS-70		0	Na	D	Options for controlling access of host computer and local control panels to operating, tuning and alarm parameters of the block.
10	STATUS_OPTS	Bitstring(2)	See Block Options	0	Na	S / O/S	See Block Options
11	IN	DS-65				D	The primary input value of the block, required for blocks that filter the input to get the PV.
12	FOLLOW	DS-66	True=follow		Na	D	Tracking input, when true causes the output to track the input.
13	LAG_TIME	DS-65			Sec	D	Specifies the lag time constant for the block. Based on a step change to the input this is the time to reach 63.2% of the final value.
14	LEAD_TIME	DS-65			Sec	D	Specifies the lead time constant applied to the input parameter.
15	BAL_TIME	Float	Positive	0	Sec	S	This specifies the time for the internal working value of bias or ratio to return to the operator set bias or ratio, in seconds.
16	OUTAGE_LIM	Float	Positive	0	Sec	S	The maximum tolerated duration for power failure. This feature is not supported.
17	UPDATE_EVT	DS-73			na	D	This alert is generated by any change to the static data.

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
18	BLOCK_ALM	DS-72			na	D	The block alarm is used for all configuration, hardware, connection failure or system problems in the block. The cause of the alert is entered in the subcode field. The first alert to become active will set the Active status in the Status attribute. As soon as the Unreported status is cleared by the alert reporting task, another block alert may be reported without clearing the Active status, if the subcode has changed.

Legend: *E* – Enumerated parameter; *na* – Adimensional parameter; *RO* – Read only; *D* – dynamic;
N – non-volatile; *S* - static

If BEHAVIOR parameter is “Adapted”:

The default value of all elements of CURVE_X is +INF.
The default value of all elements of CURVE_Y is +INF.
The required mode for writing is the actual mode, regardless the target mode: OUT.

OSDL - Output Signal Selector and Dynamic Limiter

Description

The output signal selector and dynamic limiter block (OSDL) provides two different algorithms types: As Output Selector the cascade input may be routed for one of two outputs based on the value of the OP_SELECT input parameter. The output not selected may have two ways: keeping the last value when not selected, or receive a internal value.

As Dynamic Limiter the cascade input is transferred to both output, but it is limited by the secondary inputs multiplied by a gain , plus a bias. The Dinamic Llimiter is extremely useful in one of its most important applications : combustion control with double cross limits.

The OSDL_TYPE parameter determines the algorithm used by the OSDL block. In order to change the OSDL_TYPE the block must be in Out of Service mode.

Output Signal Selector

The SP value may be controlled by an operator (Auto mode) or through a cascade control (Cas mode). In the cascade control the SP is supplied by another function block through the CAS_IN parameter. The inputs IN and IN_1 do not apply in this algorithm. It means that the block ignores the status and values of IN and IN_1 when the OSDL_TYPE is Output Selector.

The OP_SELECT is a discrete input parameter that selects one of two outputs to receive SP parameter. When the OP_SELECT is zero, the OUT parameter receives the SP parameter. Otherwise the OUT_1 parameter receives the SP parameter.

When the OP_SELECT status is not usable, the block changes to Auto, but the algorithm goes on working with the unusable value.

Most of the times, the SP is transferred to the selected output whatever is the status. Therefore, an unusable value and status in the CAS_IN will be reflected to the selected output. Only the status of upper cascade initialization will not be copied to the selected output.

Handling the non-selected output

There are two ways to handle the non-selected output, if the "Keep last value if not selected" bit in OSDL_OPTS parameter is true, the non-selected output will keep the last value. Otherwise it will receive the value contained in the NOT_SEL_VAL or NOT_SEL_VAL_1, the outputs OUT and OUT_1 respectively.

The non-selected output receives the uncertain status indicating to the downstream block that it is not selected any more. The configuration of the STATUS_OPTS in the downstream block will define how to deal with it.

IFS status propagation

If the "IFS only for selected output" bit in OSDL_OPTS parameter is true, the fault state status will be propagated only to selected output. Otherwise the status GoodCascade -IFS is propagated to both outputs, that is the default configuration.

Downstream block is not in cascade

If the downstream block of the selected output is not in cascade mode, the OSDL block goes to lman mode. And the status of BKCAL_OUT will be GoodCascade – Not Invited, it will force the upstream block to lman mode too.

If the downstream block of the non-selected output is not in cascade mode, the OSDL block will ignore it.

Dynamic Limiter

As a dynamic limiter algorithm, the outputs are the value of the "CAS_IN" parameter limited by the following values:

OUT:

High limit = $HI\_GAIN\_1 * IN\_1 + HI\_BIAS\_1$

Low limit = $LO\_GAIN\_1 * IN\_1 - LO\_BIAS\_1$

OUT_1:

High limit = $HI\_GAIN * IN + HI\_BIAS$

Low limit = $LO\_GAIN * IN - LO\_BIAS$

After the limitation, the parameters GAIN and GAIN_1 are applied as gain for the outputs OUT and OUT_1, respectively.

The normal mode of operation for the OSDL block is Cas, as well the two downstream blocks. If one downstream block is not in cascade mode, indicated by not invited status (NI) on its BKCAL_OUT, the OSDL block still continues in cascade mode. Only if both downstream blocks are not in cascade, then the OSDL changes to Iman mode and its BKCAL_OUT output changes to NI.

If the OSDL block is in Iman mode, when the cascade is initialized with a IR by a downstream block, the respective output (OUT or OUT_1) sends a IA for the downstream block and the BKCAL_OUT of the OSDL block receive the value of the respective BKCAL_IN. The OSDL block remains in IMAN mode until the downstream cascade is initialized. Then OSDL block goes to Auto mode and it sends an IR to upstream block to initialize the cascade.

After a downstream cascade initialization, the corresponding output must ramp from the last BKCAL_IN to the calculated value in BAL_TIME seconds.

The required actions as a dynamic limiter algorithm are summarized in the following table:

Mode target/actual	BKCAL_ IN	BKCAL_ IN1	BKCAL_ OUT	ACTION
Cas/Iman	NI or IR	NI or IR	NI	
Cas/Cas	NI or IR	OK	OK	BKCAL_OUT receive the CAS_IN value
Cas/Cas	OK	NI or IR	OK	BKCAL_OUT receive the CAS_IN value.
Cas/Cas	OK	OK	OK	BKCAL_OUT receive the CAS_IN value.

Legend: NI-not invited; IR-initialization request; OK-working in cascade

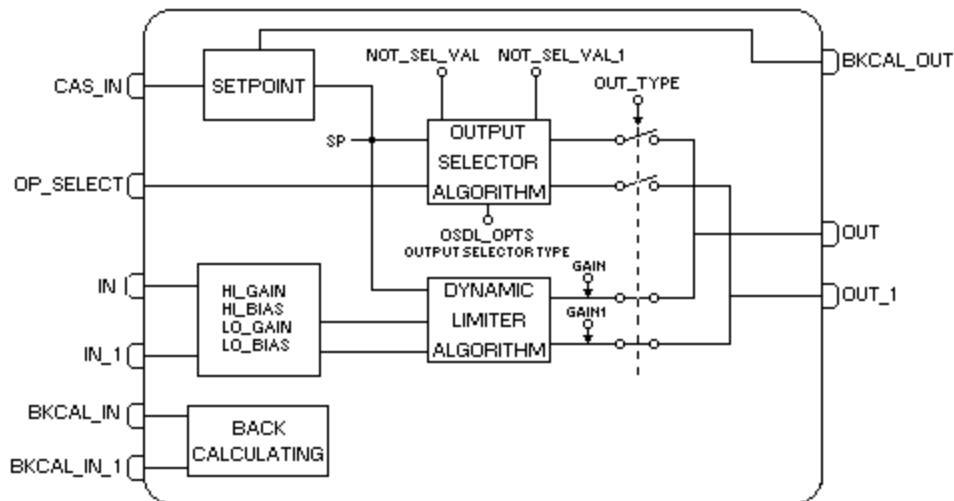
BLOCK_ERR

The BLOCK_ERR of the OSDL block will reflect the following causes:

Block Configuration Error – the configuration error occurs when the OUT_TYPE parameter has an invalid value.

Out of Service – When the block is in O/S mode.

SCHEMATIC



Supported Modes

O/S, IMAN, AUTO and CAS.

Status Handling

Standard, plus the following:

When one or both of the IN's input are Bad, special limiting action must be taken. If the "IFS if Bad IN_x " or "IFS if Bad CAS_IN " bit is true in the OSDL_OPTS parameter and the respective input is Bad, both output status go to "good IFS". If the bit is not true, the block goes to AUTO mode.

Sub-Status values received in CAS_IN will be passed onto both outputs, except for those used in the cascade initialization. An IFS will go to both the selected and the non-selected output.

Output Signal Selector And Dynamic Limiter Control Block Access

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
1	ST_REV	Unsigned16		0	None	S	
2	TAG_DESC	OctString(32)		Spaces	Na	S	
3	STRATEGY	Unsigned16		0	None	S	
4	ALERT_KEY	Unsigned8	1 to 255	0	None	S	
5	MODE_BLK	DS-69		O/S	Na	S	See Mode Parameter
6	BLOCK_ERR	Bitstring(2)			E	D	
7	CAS_IN	DS-65				D	This parameter is the remote setpoint value, which must come from another Fieldbus block, or a DCS block through a defined link.
8	SP	DS-65				N / Auto	The analog set point.
9	IN	DS-65				D	The primary input value of the block.

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
10	IN_1	DS-65				D	Numbered input parameter 1.
11	OP_SELECT	DS-66	0 thru 4		None	D	An operator adjustable parameter to force a given input to be used.
12	OUT	DS-65				D / RO	The primary analog value calculated as a result of executing the function.
13	OUT_1	DS-65				D / RO	Numbered output parameter 1.
14	GRANT_DENY	DS-70		0	Na	S / O/S	Options for controlling access of host computer and local control panels to operating, tuning and alarm parameters of the block.
15	OUT_TYPE	Unsigned8	0 : Invalid Value 1 : Output Selector 2 : Dynamic Limiter	0	E	S / Man	This parameter specifies the algorithm type that will be calculated.
16	OSDL_OPTS	Bitstring(2)		0	Na	S / O/S	Option bit string for handling the block processing.
17	HI_GAIN	Float		1.1	None	S	It is used to calculate the high limit for OUT. This gain multiplies IN_1 before adding HI_BIAS.
18	HI_BIAS	Float	Positive	0		S	It is used to calculate the high limit for OUT. This bias is added to IN_1 after multiplying by HI_GAIN.
19	LO_GAIN	Float		0.9	None	S	It is used to calculate the low limit for OUT. This gain multiplies IN_1 before subtracting LO_BIAS.
20	LO_BIAS	Float	Positive	0		S	It is used to calculate the low limit for OUT. This bias is subtracted of IN_1 after multiplying by LO_GAIN.
21	HI_GAIN_1	Float		1.1	None	S	It is used to calculate the high limit for OUT_1. This gain multiplies IN before adding HI_BIAS_1.
22	HI_BIAS_1	Float	Positive	0		S	It is used to calculate the high limit for OUT_1. This bias is added to IN after multiplying by HI_GAIN_1.
23	LO_GAIN_1	Float		0.9	None	S	It is used to calculate the low limit for OUT_1. This gain multiplies IN before subtracting LO_BIAS_1.
24	LO_BIAS_1	Float	Positive	0		S	It is used to calculate the low limit for OUT_1. This bias is subtracted of IN after multiplying by LO_GAIN_1.

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
25	GAIN	Float		0	None	S	Gain applied to OUT after limiting.
26	GAIN_1	Float		0	None	S	Gain applied to OUT_1 after limiting.
27	BKCAL_IN	DS-65				N	The value and status from a lower block's BKCAL_OUT that is used to prevent reset windup and to initialize the control loop.
28	BKCAL_IN_1	DS-65				N	The back calculated input required to initialize a lower cascade 1.
29	BKCAL_OUT	DS-66				D	The value and status required by an upper block's BKCAL_IN so that the upper block may prevent reset windup and provide bumpless transfer to closed loop control.
30	BAL_TIME	Float		0	Sec	S	This specifies the time for the internal working value of bias or ratio to return to the operator set bias or ratio, in seconds.
27	NOT_SEL_VAL	Float		0		S	Contained parameter that will set the respective output when OUT is not selected by the OP_SELECT.
28	NOT_SEL_VAL_1	Float		0		S	Contained parameter that will set the respective output when OUT_1 is not selected by the OP_SELECT.
33	UPDATE_EVT	DS-73			na	D	This alert is generated by any change to the static data.
34	BLOCK_ALM	DS-72			na	D	The block alarm is used for all configuration, hardware, connection failure or system problems in the block. The cause of the alert is entered in the subcode field. The first alert to become active will set the Active status in the Status attribute. As soon as the Unreported status is cleared by the alert reporting task, another block alert may be reported without clearing the Active status, if the subcode has changed.

Legend: E – Enumerated parameter; na – Admensional parameter; RO – Read only; D – dynamic;
N – non-volatile; S - static

If BEHAVIOR parameter is “Adapted”:

The default value of OUT_TYPE is “Dynamic limiter”.

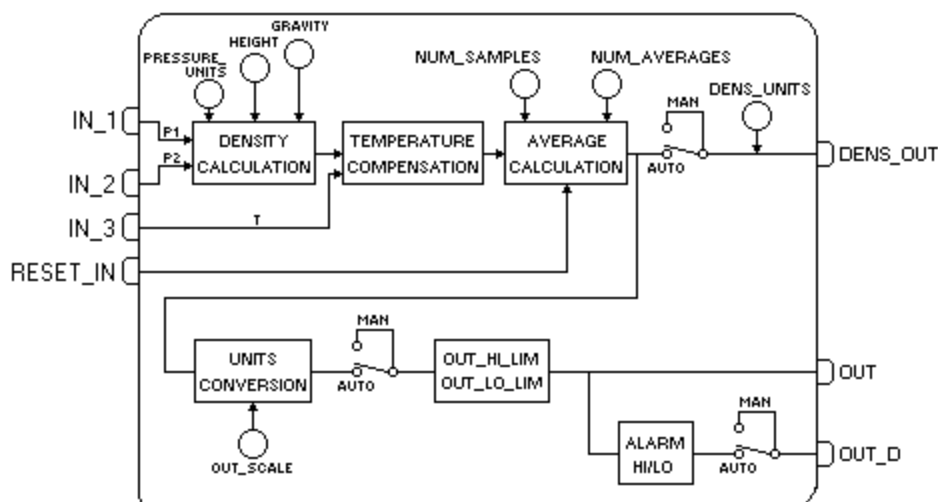
The required mode for writing is the actual mode, regardless the target mode: SP.

DENS - Density

Overview

This function block has an algorithm to calculate density in different kinds of engineering units, as Plato degree, Brix, TC and INPM.

Schematic



Description

The algorithm to calculate the density is based on the pressure in two points of the tank in a known difference height.

The temperature is compensated in the density calculation. The DENS_OUT parameter is the compensated density in g/cm^3 . The OUT parameter is the compensated density in a different engineering unit for chosen by the EU_SEL parameter.

The density block provides alarm condition and the discrete alarm output to be used any block. If the density exceeds HI_LIM or LO_LIM an alarm is indicated in HI_ALM or LO_ALM and the output OUT_D will be set to true. If one or both of the limit parameters are set +/- INF this indicate that this alarm is disabled.

The RESET_IN is a discrete input, which on a false-to-true transition resets the density block. Including the previous density, the round buffer and all outputs of the block. The status will be the same as the last cycle.

BLOCK_ERR

The BLOCK_ERR of the DENS block will reflect the following causes:
Out of Service – it occurs when the block is in O/S mode.

Supported Modes

O/S, Man and Auto.

Mode Handling

Manual mode disconnects the outputs from the algorithm and permits manual substitution of the OUT, DENS_OUT, OUT_D values for test or other purposes. Although the OUT_D parameter is also disconnected from the alarm, the alarm and the limits (OUT_HI_LIM and OUT_LO_LIM) continue to check the output.

Status Handling

The primary input status (IN_1 and IN_2 parameters) are propagated to the outputs. If the status of any primary input becomes bad or uncertain and their respective option "use uncertain" of STATUS_OPTS is not set, the actual mode block will be forced to manual and the algorithm stops the calculation.

If the secondary input (IN_3 parameter) is unusable, the algorithm uses the last usable value and the output status will be Uncertain.

A bad status in RESET_IN input does not stop the algorithm.
If target mode is Man then the output status is Good.

Density Block Access

Index	Parameter	DataType (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
7	ST_REV	Unsigned16		0	None	S	
8	TAG_DESC	OctString(32)		Spaces	Na	S	
9	STRATEGY	Unsigned16		0	None	S	
4	ALERT_KEY	Unsigned8	1 to 255	0	None	S	
5	MODE_BLK	DS-69		O/S	Na	S	See Mode Parameter
6	BLOCK_ERR	Bitstring(2)			E	D / RO	
7	OUT	DS-65	OUT_SCALE +/- 10%		OUT	D / Man	The analog value calculated as a result of executing the function.
8	OUT_SCALE	DS-68		0-100%	OUT	S / Man	The high and low scale values to the OUT parameter.
9	EU_SEL	Unsigned8	0:Plato degree 1:Brix 2:TC 3:INPM	0	E	S / Man	Selection of engineering unit for density.
10	GRANT_DENY	DS-70		0	na	D	
11	STATUS_OPTS	Bitstring(2)	See Block Options	0	Na	S / O/S	See Block Options
12	IN_1	DS-65				D	The first input (pressure P1) must have the same engineering units of IN_2.

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
13	IN_2	DS-65				D	The second input (pressure P2) must have the same engineering units of IN_1.
14	IN_3	DS-65				D	The third input. (Temperature T in Degrees Celsius)
15	PRESSURE_UNITS	Unsigned16	See valid pressure units	1144 (g/cm ²)	E	S	This is the pressure engineering unit of IN_1 and IN_2.
16	HEIGHT	Float	Positive	1000.0	mm	S	Distance between the two pressure transmitters. The engineering units must be compatible to the inputs IN_1 and IN_2. If it is mmH2O, the EU of HEIGHT is mm.
17	GRAVITY	Float	Positive	9.80665	m/s ²	S	The gravity acceleration used in the density calculation, the EU of GRAVITY is m/s ² .
18	NUM_SAMPLES	Unsigned16	1-1000	10	Na	S	Number of samples.
19	NUM_AVERAGES	Unsigned16	1-30	10	Na	S	Number of averages in the round buffer.
20	DENS_OUT	DS-65			g/cm ³	D / Man	The density compensated by the temperature.
21	DENS_UNITS	Unsigned16		1100	g/cm ³	S / RO	This is the engineering unit of DENS_OUT that is fixed in g/cm ³ .
22	OUT_D	DS-66				D	This is a discrete output parameter to indicate alarm state.
23	RESET_IN	DS-66				D	When it is true, it resets the average calculation and it clears the round buffer of averages.
24	OUT_HI_LIM	Float		100	OUT	S	High limit for OUT.
25	OUT_LO_LIM	Float		0	OUT	S	Low limit for OUT.
26	UPDATE_EVT	DS-73			Na	D	This alert is generated by any change to the static data.
27	BLOCK_ALM	DS-72			Na	D	The block alarm is used for all configuration, hardware, connection failure or system problems in the block. The cause of the alert is entered in the subcode field. The first alert to become active will set the Active status in the Status attribute. As soon as the Unreported status is cleared by the alert reporting task, another block alert may be reported without clearing the Active status, if the subcode has changed.
28	ALARM_SUM	DS-74			Na	S	The current alert status, unacknowledged states, unreported states, and disabled states of the alarms associated with the function block.

Index	Parameter	DataType (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
29	ACK_OPTION	Bitstring(2)	0: Auto ACK Disable 1: Auto ACK Enable	0	Na	S	Selection of whether alarms associated with the block will be automatically acknowledged
30	ALARM_HYS	Float	0 to 50 %	0.5%	%	S	Alarm hysteresis parameter. In order to clear the alarm the amount the PV must return within the alarm limit plus hysteresis.
31	HI_PRI	Unsigned8	0 to 15			S	Priority of the high alarm.
32	HI_LIM	Float	OUT_SCALE, +INF	+INF	OUT	S	The setting for high alarm in engineering units.
33	LO_PRI	Unsigned8	0 to 15			S	Priority of the low alarm.
34	LO_LIM	Float	OUT_SCALE, - INF	-INF	OUT	S	The setting for low alarm in engineering units.
35	HI_ALM	DS-71			OUT	D	The status for high alarm and its associated time stamp.
36	LO_ALM	DS-71			OUT	D	The status for low alarm and its associated time stamp.

Legend: E – Enumerated parameter; na – Adimensional parameter; RO – Read only; D – dynamic;
N – non-volatile; S - static

If BEHAVIOR parameter is “Adapted”:

The required mode for writing is the actual mode, regardless the target mode : OUT, DENS_OUT,

Valid Pressure Units

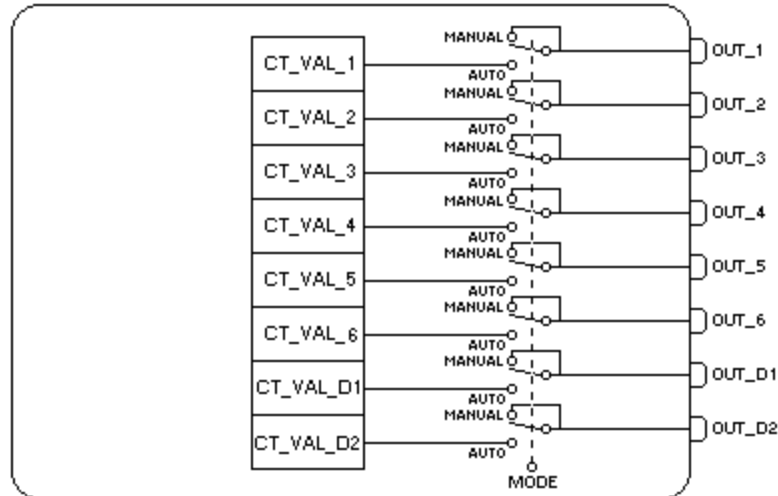
Index	Unit
1130	PA
1133	KPA
1132	MPA
1137	BAR
1138	MBAR
1139	TORR
1140	ATM
1141	PSI
1144	GCM2
1145	KGCM2
1148	INH20
1147	INH204C
1151	MMH20
1150	MMH204C
1154	FTH20
1156	INHG
1158	MMHG

CT – Constant

Overview

The constant function block generates constant values to use in inputs parameters of other blocks.

Schematic



Description

The constant function block has 6 analog constants and 2 discrete constants to connect in any other blocks.

If the mode is Man then is permitted manual substitution of all output values. In Auto mode the output values are the respective constant values.

Supported modes

O/S, MAN, AUTO

Status handling

Parameters

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
1	ST_REV	Unsigned16		0	None	S	
2	TAG_DESC	OctString(32)		Spaces	Na	S	
3	STRATEGY	Unsigned16		0	None	S	
4	ALERT_KEY	Unsigned8	1 to 255	0	None	S	
5	MODE_BLK	DS-69		O/S	Na	S	See Mode Parameter
6	BLOCK_ERR	Bitstring(2)			E	D	

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
7	OUT_1	DS-65				N / Man	Numbered output 1.
8	OUT_2	DS-65				D / Man	Numbered output 2.
9	OUT_3	DS-65				D / Man	Numbered output 3.
10	OUT_4	DS-65				D / Man	Numbered output 4.
11	OUT_5	DS-65				D / Man	Numbered output 5.
12	OUT_6	DS-65				D / Man	Numbered output 6.
13	OUT_D1	DS-66				N / Man	Numbered discrete output 1.
14	OUT_D2	DS-66				D / Man	Numbered discrete output 2.
15	CT_VAL_1	Float		0		S	Analog constant value transferred to the output OUT_1.
16	CT_VAL_2	Float		0		S	Analog constant value transferred to the output OUT_2.
17	CT_VAL_3	Float		0		S	Analog constant value transferred to the output OUT_3.
18	CT_VAL_4	Float		0		S	Analog constant value transferred to the output OUT_4.
19	CT_VAL_5	Float		0		S	Analog constant value transferred to the output OUT_5.
20	CT_VAL_6	Float		0		S	Analog constant value transferred to the output OUT_6.
21	CT_VAL_D1	Unsigned8		0		S	Discrete constant value transferred to the output OUT_D1.
22	CT_VAL_D2	Unsigned8		0		S	Discrete constant value transferred to the output OUT_D2.
23	UPDATE_EVT	DS-73			Na	D	This alert is generated by any change to the static data.
24	BLOCK_ALM	DS-72			Na	D	The block alarm is used for all configuration, hardware, connection failure or system problems in the block. The cause of the alert is entered in the subcode field. The first alert to become active will set the Active status in the Status attribute. As soon as the Unreported status is cleared by the alert reporting task, another block alert may be reported without clearing the Active status, if the subcode has changed.

Legend: E – Enumerated parameter; na – Adimensional parameter; RO – Read only; D – dynamic;
N – non-volatile; S - static

If BEHAVIOR parameter is “Adapted”:

The required mode for writing is the actual mode, regardless the target mode: OUT_1, OUT_2, OUT_3, OUT_4, OUT_5, OUT_6, OUT_D1 and OUT_D2.

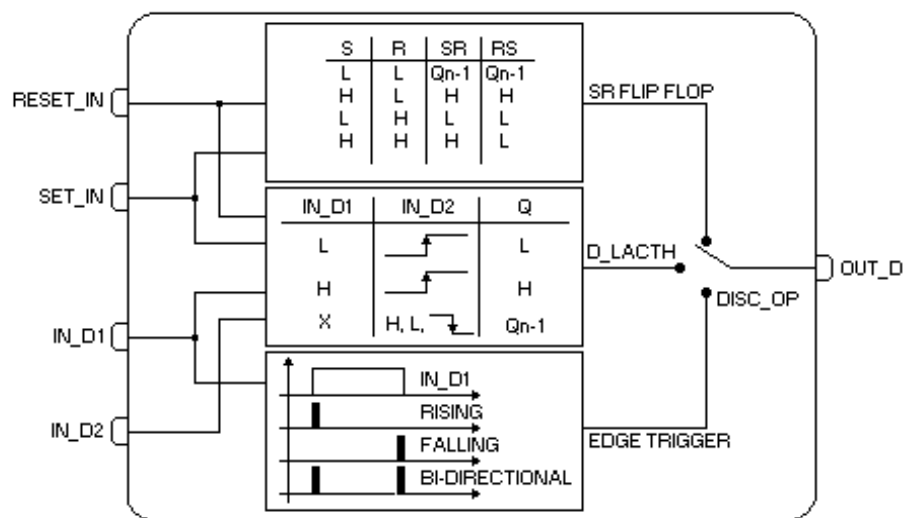
FFET - Flip-Flop and Edge Trigger

Overview

It can be configured to work as :

SR flip-flop
 RS flip-flop
 D-latch
 Rising edge trigger
 Falling edge trigger
 Bi-directional edge trigger

Schematic



Description

The following tables summarizes the block behavior:

RESET_IN	SET_IN	OUT_D (SR flip-flop)	OUT_D (RS flip-flop)
L	L	Q_{n-1}	Q_{n-1}
H	L	L	L
L	H	H	H
H	H	H	L

RESET_IN	SET_IN	IN_D1	IN_D2	OUT_D (D-latch)
L	H	X	X	H
H	L	X	X	L
H	H	X	X	H
L	L	Rising	L	L
L	L	Rising	H	H
L	L	H,L or Falling	X	OUT_D _{n-1}

RESET_IN	SET_IN	IN_D1	OUT_D (Rising Edge)	OUT_D (Falling Edge)	OUT_D (Bi-directional)
L	L	Rising	H	L	H
L	L	Falling	L	H	H
L	L	No transition	L	L	L
X	H	X	H	H	H
H	L	X	L	L	L

BLOCK_ERR

The BLOCK_ERR of the FFET block will reflect the following causes:
Out of Service – When the block is in O/S mode.

Supported Modes

O/S, MAN and AUTO.

Status Handling

Parameters

Index	Parameter	DataType (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
10	ST_REV	Unsigned16		0	None	S	
11	TAG_DESC	OctString(32)		Spaces	Na	S	
12	STRATEGY	Unsigned16		0	None	S	
4	ALERT_KEY	Unsigned8	1 to 255	0	None	S	
5	MODE_BLK	DS-69		O/S	Na	S	See Mode Parameter
6	BLOCK_ERR	Bitstring(2)			E	D / RO	

Index	Parameter	DataType (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
7	DISC_OP	Unsigned8	0 : SR flip-flop 1 : RS flip-flop 2 : D-latch 3 : rising edge 4 : falling edge 5 : bi-directional edge		E	S / Man	Selection of discrete operation.
8	STATUS_OPTS	Bitstring(2)				S / O/S	
9	IN_D1	DS-66				D	Numbered discrete input 1.
10	IN_D2	DS-66				D	Numbered discrete input 2.
11	SET_IN	DS-66	0 : Off 1 : set			D	The set input.
12	RESET_IN	DS-66	0 : Off 1 : reset			D	The reset input.
13	OUT_D	DS-66				N / Man	The output of flip-flop.
14	UPDATE_EVT	DS-73			Na	D	This alert is generated by any change to the static data.
15	BLOCK_ALM	DS-72			Na	D	The block alarm is used for all configuration, hardware, connection failure or system problems in the block. The cause of the alert is entered in the subcode field. The first alert to become active will set the Active status in the Status attribute. As soon as the Unreported status is cleared by the alert reporting task, another block alert may be reported without clearing the Active status, if the subcode has changed.

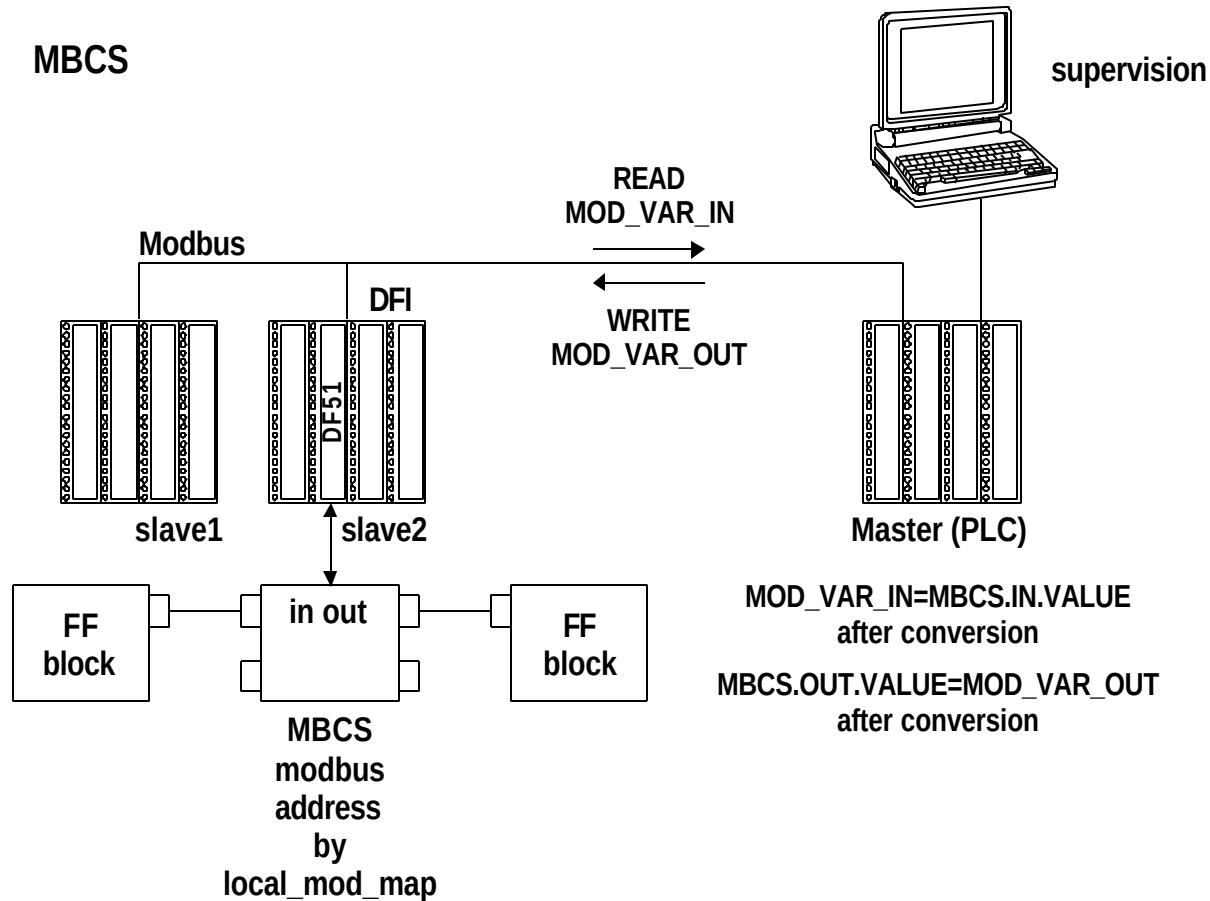
Legend: E – Enumerated parameter; na – Adimensional parameter; RO – Read only; D – dynamic;
N – non-volatile; S - static

If BEHAVIOR parameter is “Adapted”:

The required mode for writing is the actual mode, regardless the target mode : OUT_D

MBCS – ModBus Control Slave

Overview



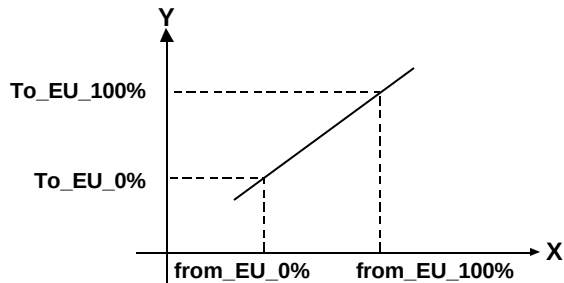
Procedure to handle the conversion from FF parameter to Modbus variable:

Load INx_VALUE.
 Calculate $Y = A * INx\_VALUE + B$.
 Convert Y to DATA_TYPE_IN, generating MOD_VAR_IN.
 Store MOD_VAR_IN.

Procedure to handle the conversion from Modbus variable to FF parameter:

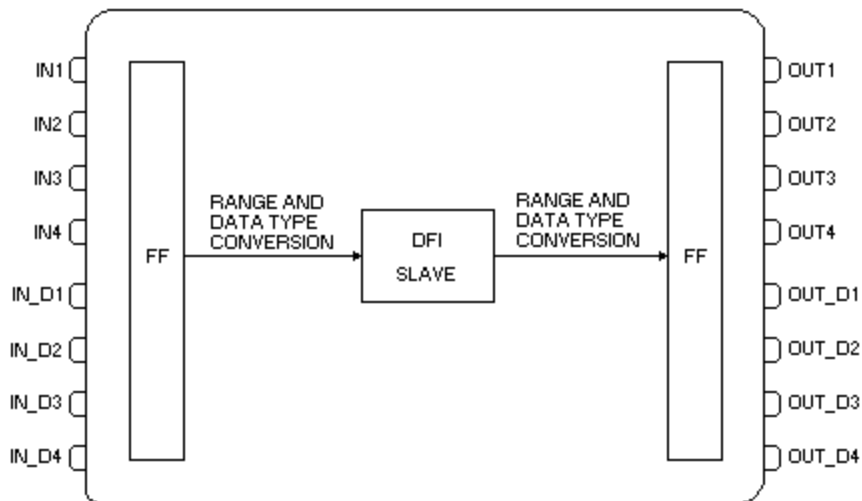
Load MOD_VAR_OUT.
 Convert MOD_VAR_OUT to float, generating Y
 Calculate $OUTx\_VALUE = (A * Y + B)$.
 Store OUTx_VALUE.

$A = (To\_EU\_100\% - To\_EU\_0\%) / (From\_EU\_100\% - From\_EU\_0\%)$
 $B = To\_EU\_0\% - A * From\_EU\_0\%;$



INx_VALUE, OUTx_VALUE: FF parameters
MOD_VAR_IN, MOD_VAR_OUT: Modbus variables
Y: auxiliary float variable

Schematic



BLOCK_ERR

The BLOCK_ERR of the MBCS block will reflect the following causes:

Other: it occurs when the conversion from Y to DATA_TYPE_IN results in a value out of range of this data type.

Out of Service: it occurs when the block is in O/S mode.

Parameters

Index	Parameter	DataType (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
1	ST_REV	Unsigned16		0	None	S	
2	TAG_DESC	OctString(32)		Spaces	Na	S	
3	STRATEGY	Unsigned16		0	None	S	
4	ALERT_KEY	Unsigned8	1 to 255	0	None	S	
5	MODE_BLK	DS-69		O/S	Na	S	See Mode Parameter
6	BLOCK_ERR	Bitstring(2)			E	D / RO	
7	LOCAL_MOD_MAP	Unsigned8	0 to 15	0		S / O/S	Define the modbus addresses.
8	IN1	DS-65				N	Analog input 1

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
9	SCALE_CONV_IN1	DS-256				S / M	Information to generate constants A and B em equation $Y=A*X+B$.
10	IN2	DS-65				N	Analog input 2
11	SCALE_CONV_IN2	DS-256				S / M	Information to generate constants A and B em equation $Y=A*X+B$.
12	IN3	DS-65				N	Analog input 3
13	SCALE_CONV_IN3	DS-256				S / M	Information to generate constants A and B em equation $Y=A*X+B$.
14	IN4	DS-65				N	Analog input 4
15	SCALE_CONV_IN4	DS-256				S / M	Information to generate constants A and B em equation $Y=A*X+B$.
16	IN_D1	DS-66				N	Discrete input 1
17	IN_D2	DS-66				N	Discrete input 2
18	IN_D3	DS-66				N	Discrete input 3
19	IN_D4	DS-66				N	Discrete input 4
20	OUT1	DS-65				N / Man	Analog output 1
21	SCALE_CONV_OUT1	DS-257				S / M	Information to generate constants A and B em equation $Y=A*X+B$ plus output status.
22	OUT2	DS-65				N / Man	Analog output 2
23	SCALE_CONV_OUT2	DS-257				S / M	Information to generate constants A and B em equation $Y=A*X+B$ plus output status.
24	OUT3	DS-65				N / Man	Analog output 3
25	SCALE_CONV_OUT3	DS-257				S / M	Information to generate constants A and B em equation $Y=A*X+B$ plus output status.
26	OUT4	DS-65				N / Man	Analog output 4
27	SCALE_CONV_OUT4	DS-257				S / M	Information to generate constants A and B em equation $Y=A*X+B$ plus output status.
28	OUT_D1	DS-66				N / Man	Discrete output 1
29	STATUS_OUT_D1	Unsigned8				S / M	Status to OUT_D1 if master will not update.
30	OUT_D2	DS-66				N / Man	Discrete output 2
31	STATUS_OUT_D2	Unsigned8				S / M	Status to OUT_D2 if master will not update.
32	OUT_D3	DS-66				N / Man	Discrete output 3
33	STATUS_OUT_D3	Unsigned8				S / M	Status to OUT_D3 if master will not update.
34	OUT_D4	DS-66				N / Man	Discrete output 4
35	STATUS_OUT_D4	Unsigned8				S / M	Status to OUT_D4 if master will not update.
36	UPDATE_EVT	DS-73			Na	D	This alert is generated by any change to the static data.
37	BLOCK_ALM	DS-72			Na	D	The block alarm is used for all configuration, hardware, connection failure or system problems in the block. The cause of the alert is entered in the subcode field. The first alert to become active will set the Active status in the Status attribute. As soon as the Unreported status is cleared by the alert reporting task, another block alert may be reported without clearing the Active status, if the subcode has changed.

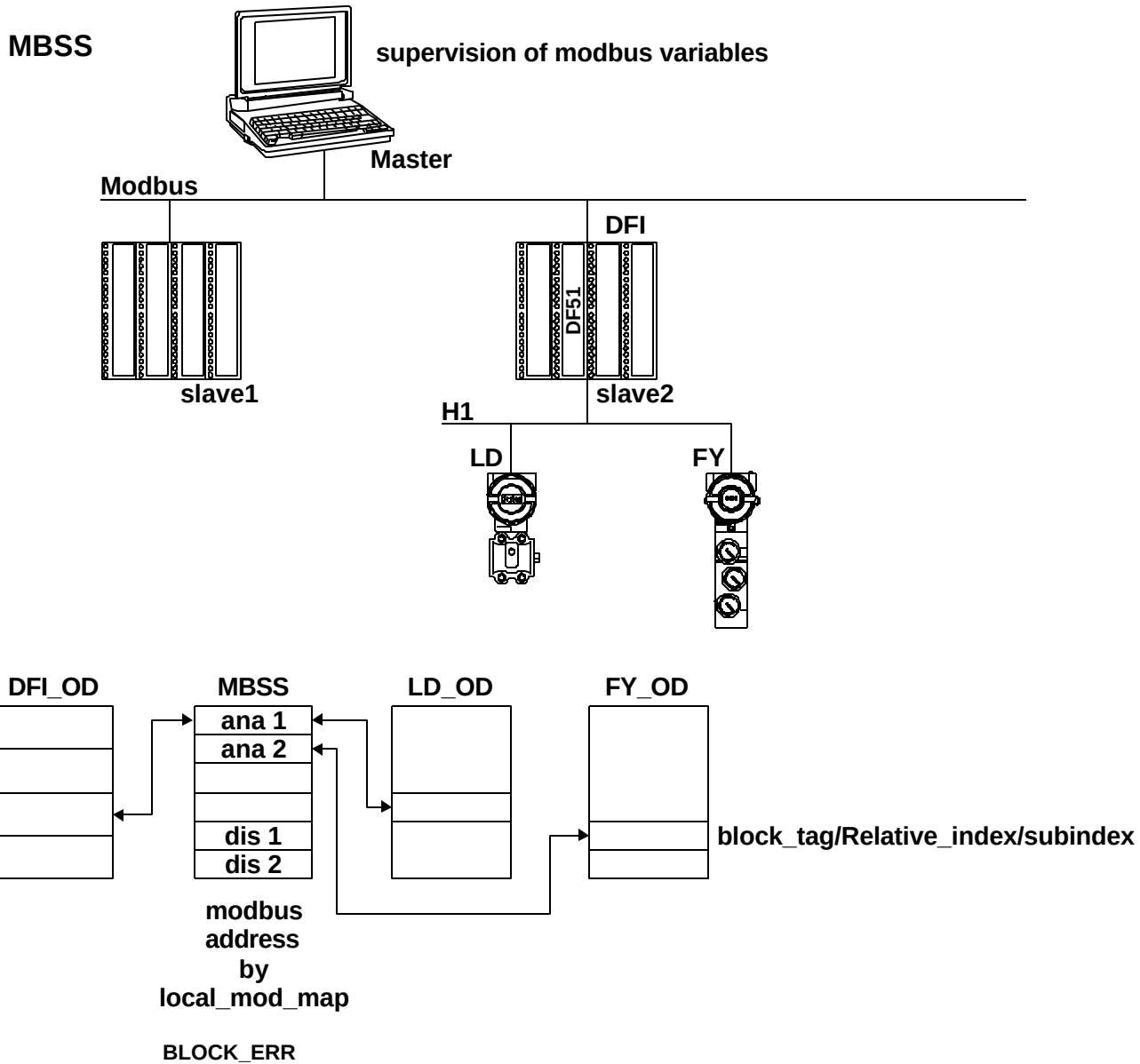
Legend: E – Enumerated parameter; na – Adimensional parameter; RO – Read only; D – dynamic; N – non-volatile; S - static

Remarks:

If the OUTs aren't updated by Modbus Master in a time specified by user (parameter TIMEOUT in MBCF), it will be generated a "bad status".
If $\text{TIMEOUT} < \text{Macrocycle}$, $\text{TIMEOUT} = \text{Macrocycle}$.

MBSS – ModBus Supervision Slave

Overview



The BLOCK_ERR of the MBSS block will reflect the following causes:

Block Configuration Error: If it is requested a tag with a data type different from permitted or invalid or not found block tag.

Out of Service: it occurs when the block is in O/S mode.

Parameters

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
1	ST_REV	Unsigned16		0	None	S	
2	TAG_DESC	OctString(32)		Spaces	Na	S	
3	STRATEGY	Unsigned16		0	None	S	
4	ALERT_KEY	Unsigned8	1 to 255	0	None	S	
5	MODE_BLK	DS-69		O/S	Na	S	See Mode Parameter
6	BLOCK_ERR	Bitstring(2)			E	D / RO	
7	LOCAL_MOD_MAP	Unsigned8	0 to 15	0		S / O/S	Define the modbus addresses.
8	F_ID1	DS-262				S / O/S	Information to locate float parameter.
9	FVALUE1	Float		0		N	Value from requested float parameter.
10	F_ID2	DS-262				S / O/S	Information to locate float parameter.
11	FVALUE2	Float		0		N	Value from requested float parameter.
12	F_ID3	DS-262				S / O/S	Information to locate float parameter.
13	FVALUE3	Float		0		N	Value from requested float parameter.
14	F_ID4	DS-262				S / O/S	Information to locate float parameter.
15	FVALUE4	Float		0		N	Value from requested float parameter.
16	F_ID5	DS-262				S / O/S	Information to locate float parameter.
17	FVALUE5	Float		0		N	Value from requested float parameter.
18	F_ID6	DS-262				S / O/S	Information to locate float parameter.
19	FVALUE6	Float		0		N	Value from requested float parameter.
20	F_ID7	DS-262				S / O/S	Information to locate float parameter.
21	FVALUE7	Float		0		N	Value from requested float parameter.
22	F_ID8	DS-262				S / O/S	Information to locate float parameter.
23	FVALUE8	Float		0		N	Value from requested float parameter.
24	I_ID1	DS-262				S / O/S	Information to locate integer parameter.
25	IVALUE1	Integer32		0		N	Value from requested integer parameter.
26	I_ID2	DS-262				S / O/S	Information to locate integer parameter.
27	IVALUE2	Integer32		0		N	Value from requested integer parameter.
28	I_ID3	DS-262				S / O/S	Information to locate integer parameter.
29	IVALUE3	Integer32		0		N	Value from requested integer parameter.
30	I_ID4	DS-262				S / O/S	Information to locate integer parameter.
31	IVALUE4	Integer32		0		N	Value from requested integer parameter.
32	B_ID1	DS-262				S / O/S	Information to locate boolean parameter.
33	BVALUE1	Boolean		TRUE		N	Value from requested boolean parameter.
34	B_ID2	DS-262				S / O/S	Information to locate boolean parameter.
35	BVALUE2	Boolean		TRUE		N	Value from requested boolean parameter.
36	B_ID3	DS-262				S / O/S	Information to locate boolean parameter.
37	BVALUE3	Boolean		TRUE		N	Value from requested boolean parameter.
38	B_ID4	DS-262				S / O/S	Information to locate boolean parameter.
39	BVALUE4	Boolean		TRUE		N	Value from requested boolean parameter.
40	UPDATE_EVT	DS-73			Na	D	This alert is generated by any change to the static data.
41	BLOCK_ALM	DS-72			Na	D	The block alarm is used for all configuration, hardware, connection failure or system problems in the block. The cause of the alert is entered in the subcode field. The first alert to become active will set the Active status in the Status attribute. As soon as the Unreported status is cleared by the alert reporting task, another block alert may be reported without clearing the Active status, if the subcode has changed.

Legend: E – Enumerated parameter; na – Adimensional parameter; RO – Read only; D – dynamic; N – non-volatile; S – static

Remarks:

BVALUEx parameters can address FF block parameters of the following data types: boolean, integer8 and unsigned8. Those data types are automatically converted to bit (0 or 1) and vice versa for Modbus supervision and also converted to boolean parameter (BVALUEx).

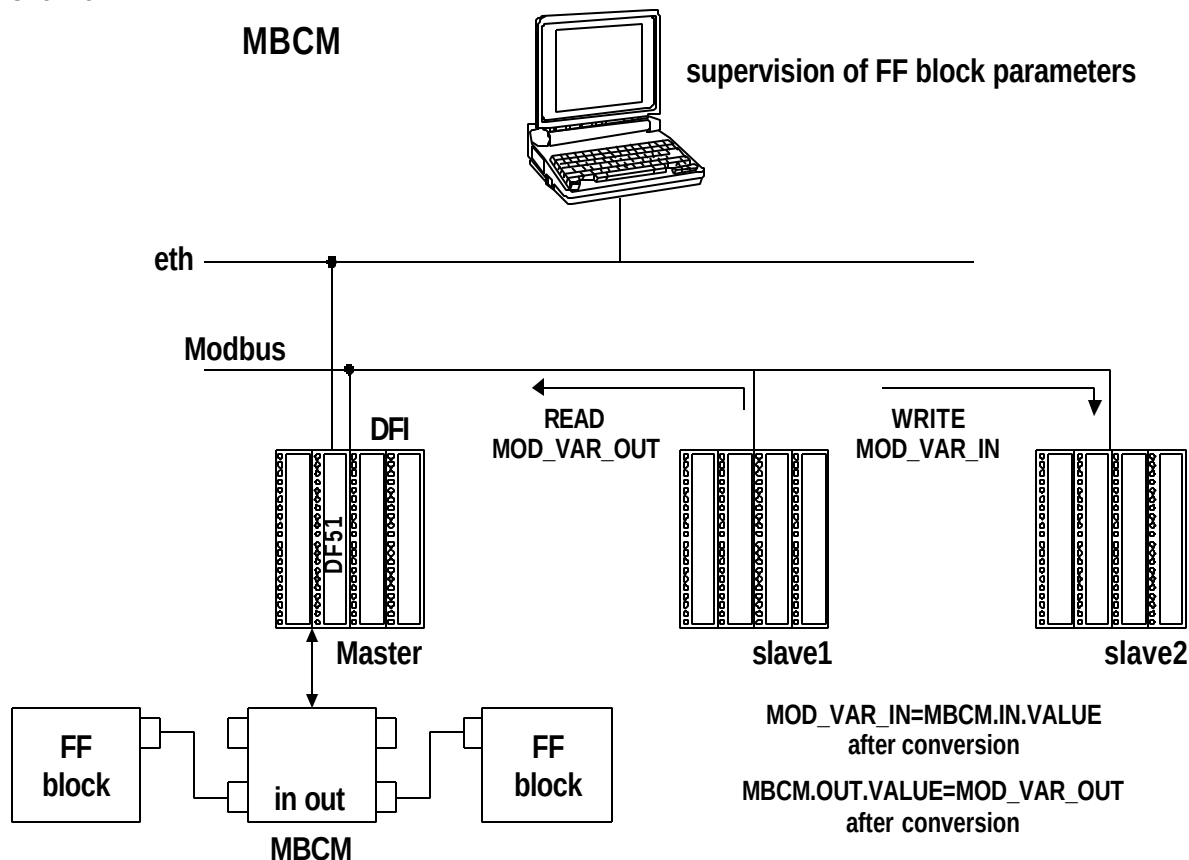
IVALEUx parameters can address FF block parameters of the following data types: Integer8, Integer16, Integer32, Unsigned8, Unsigned16 and Unsigned32.

Each analog parameter (IVALEUx) is mapping as two analog registers in Modbus, ie, four bytes. Thus, when addressing a FF block parameter with one or two bytes, such parameter will be promoted to Unsigned32 or Integer32.

If Relative Index = 5 (MODE_BLK) e Sub Index = 0, it is performed a writing in Sub Index 1 and a reading in Sub Index 2.

MBCM – ModBus Control Master

Overview



Procedure to handle the conversion from FF parameter to Modbus variable:

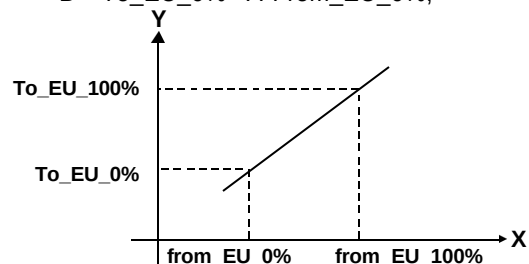
Load INx_VALUE.
 Calculate $Y = (A * INx\_VALUE + B)$.
 Convert Y to DATA_TYPE_IN, generating MOD_VAR_IN.
 Write MOD_VAR_IN.

Procedure to handle the conversion from Modbus variable to FF parameter:

Read MOD_VAR_OUT.
 Convert MOD_VAR_OUT to float, generating Y
 Calculate $OUTx\_VALUE = (A * Y + B)$.
 Store OUTx_VALUE.

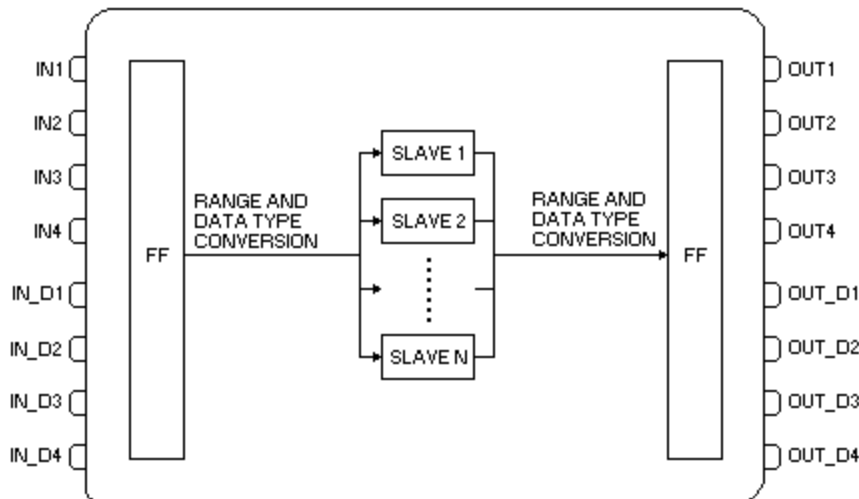
$$A = (To\_EU\_100\% - To\_EU\_0\%) / (From\_EU\_100\% - From\_EU\_0\%)$$

$$B = To\_EU\_0\% - A * From\_EU\_0\%;$$



IN_VALUE, OUT_VALUE: FF parameters
MOD_VAR_IN, MOD_VAR_OUT: Modbus variables
Y: auxiliary float variable

Schematic



BLOCK_ERR

The BLOCK_ERR of the MBCM block will reflect the following causes:
Other: it occurs when the conversion from Y to DATA_TYPE_IN results in a value out of range of this data type.
Out of Service: it occurs when the block is in O/S mode.

Parameters

Index	Parameter	DataType (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
1	ST_REV	Unsigned16		0	None	S	
2	TAG_DESC	OctString(32)		Spaces	Na	S	
3	STRATEGY	Unsigned16		0	None	S	
4	ALERT_KEY	Unsigned8	1 to 255	0	None	S	
5	MODE_BLK	DS-69		O/S	Na	S	See Mode Parameter
6	BLOCK_ERR	Bitstring(2)			E	D / RO	
7	LOCAL_MOD_MAP	Unsigned8	0 to 15	0		S / O/S	Define the modbus addresses.
8	COMM_STATUS	Bitstring(2)		0	E	D / RO	Indicate if communication from slave is good or not (each bit corresponds to a Modbus variable).
9	IN1	DS-65				N	Analog input 1
10	SCALE_LOC_IN1	DS-259				S / M	Information to generate constants A and B in equation $Y=A*X+B$ plus the addresses in a slave device.
11	IN2	DS-65				N	Analog input 2
12	SCALE_LOC_IN2	DS-259				S / M	Information to generate constants A and B in equation $Y=A*X+B$ plus the addresses in a slave device.
13	IN3	DS-65				N	Analog input 3
14	SCALE_LOC_IN3	DS-259				S / M	Information to generate constants A and B in equation $Y=A*X+B$ plus the addresses in a slave device.

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Unit s	Store / Mode	Description
15	IN4	DS-65				N	Analog input 4
16	SCALE_LOC_IN4	DS-259				S / M	Information to generate constants A and B em equation $Y=A*X+B$ plus the addresses in a slave device.
17	IN_D1	DS-66				N	Discrete input 1
18	LOCATOR_IN_D1	DS-261				S / O/S	Addresses in a slave device.
19	IN_D2	DS-66				N	Discrete input 2
20	LOCATOR_IN_D2	DS-261				S / O/S	Addresses in a slave device.
21	IN_D3	DS-66				N	Discrete input 3
22	LOCATOR_IN_D3	DS-261				S / O/S	Addresses in a slave device.
23	IN_D4	DS-66				N	Discrete input 4
24	LOCATOR_IN_D4	DS-261				S / O/S	Addresses in a slave device.
25	OUT1	DS-65				N / Man	Analog output 1
26	SCALE_LOC_OUT1	DS-259				S / M	Information to generate constants A and B em equation $Y=A*X+B$ plus the addresses in a slave device.
27	OUT2	DS-65				N / Man	Analog output 2
28	SCALE_LOC_OUT2	DS-259				S / M	Information to generate constants A and B em equation $Y=A*X+B$ plus the addresses in a slave device.
29	OUT3	DS-65				N / Man	Analog output 3
30	SCALE_LOC_OUT3	DS-259				S / M	Information to generate constants A and B em equation $Y=A*X+B$ plus the addresses in a slave device.
31	OUT4	DS-65				N / Man	Analog output 4
32	SCALE_LOC_OUT4	DS-259				S / M	Information to generate constants A and B em equation $Y=A*X+B$ plus the addresses in a slave device.
33	OUT_D1	DS-66				N / Man	Discrete output 1
34	LOCATOR_OUT_D1	DS-261				S / O/S	Addresses in a slave device.
35	OUT2_D2	DS-66				N / Man	Discrete output 2
36	LOCATOR_OUT_D2	DS-261				S / O/S	Addresses in a slave device.
37	OUT_D3	DS-66				N / Man	Discrete output 3
38	LOCATOR_OUT_D3	DS-261				S / O/S	Addresses in a slave device.
39	OUT_D4	DS-66				N / Man	Discrete output 4
40	LOCATOR_OUT_D4	DS-261				S / O/S	Addresses in a slave device.
41	UPDATE_EVT	DS-73			Na	D	This alert is generated by any change to the static data.
42	BLOCK_ALM	DS-72			Na	D	The block alarm is used for all configuration, hardware, connection failure or system problems in the block. The cause of the alert is entered in the subcode field. The first alert to become active will set the Active status in the Status attribute. As soon as the Unreported status is cleared by the alert reporting task, another block alert may be reported without clearing the Active status, if the subcode has changed.

Legend: E – Enumerated parameter; na – Admensional parameter; RO – Read only; D – dynamic; N – non-volatile; S - static

Relation between the bits in COMM_STATUS and Modbus addresses

Bit	Variable
0	BAD COMM MOD_VAR_IN1
1	BAD COMM MOD_VAR_IN2
2	BAD COMM MOD_VAR_IN3

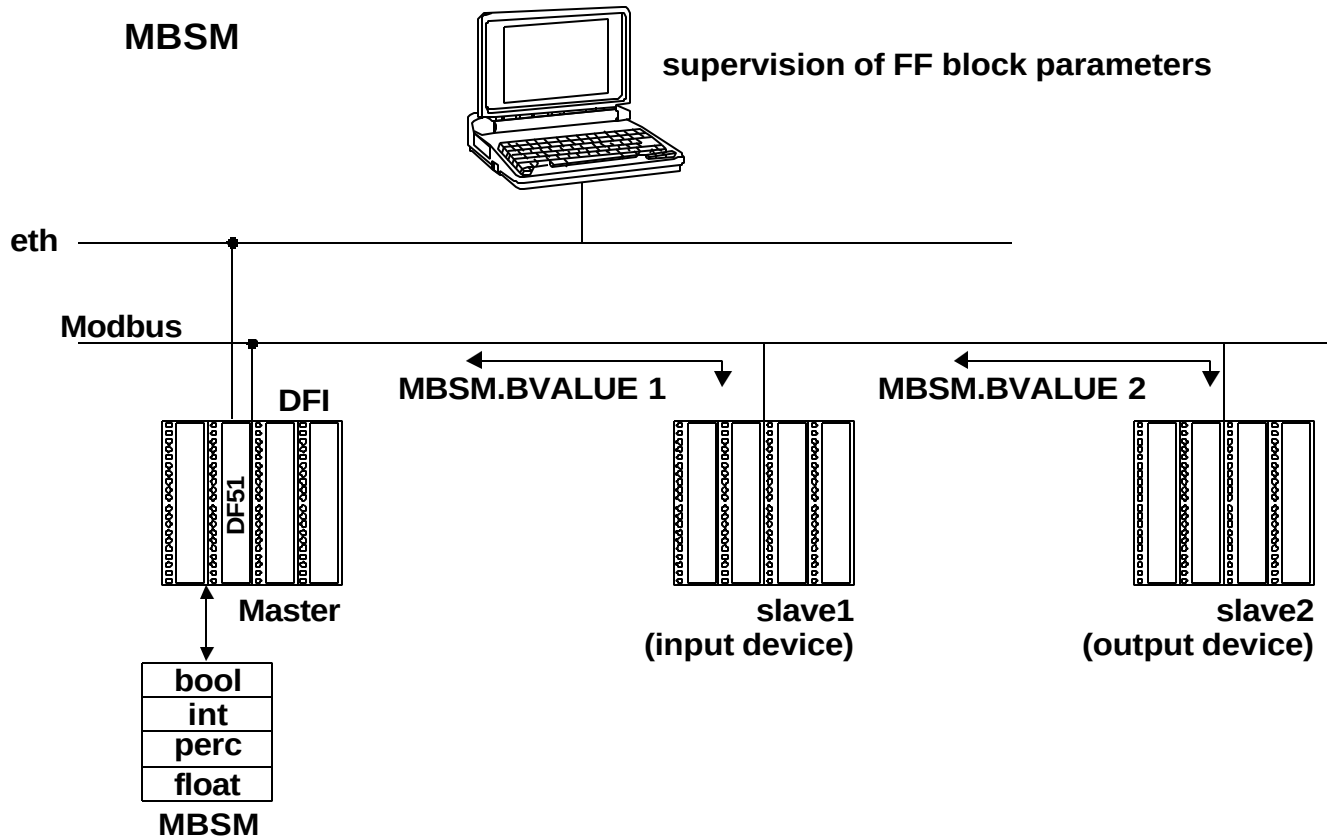
3	BAD COMM MOD VAR IN4
4	BAD COMM MOD VAR IN_D1
5	BAD COMM MOD VAR IN_D1
6	BAD COMM MOD VAR IN_D1
7	BAD COMM MOD VAR IN_D1
8	BAD COMM MOD VAR_OUT1
9	BAD COMM MOD VAR_OUT1
10	BAD COMM MOD VAR_OUT1
11	BAD COMM MOD VAR_OUT1
12	BAD COMM MOD VAR_OUT_D1
13	BAD COMM MOD VAR_OUT_D1
14	BAD COMM MOD VAR_OUT_D1
15	BAD COMM MOD VAR_OUT_D1

Remarks:

Each bit corresponds to an OR between the value and status, indicating if communication with slave is good or bad.
 If it is only used the value, the status is considered zero.
 If it is only used the status, the value is considered zero.

MBSM – ModBus Supervision Master

Overview



Parameters

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
1	ST_REV	Unsigned16		0	None	S	
2	TAG_DESC	OctString(32)		Spaces	Na	S	
3	STRATEGY	Unsigned16		0	None	S	
4	ALERT_KEY	Unsigned8	1 to 255	0	None	S	
5	MODE_BLK	DS-69		O/S	Na	S	See Mode Parameter
6	BLOCK_ERR	Bitstring(2)			E	D / RO	
7	LOCAL_MOD_MAP	Unsigned8	0 to 15	0		S / O/S	Define the modbus addresses.
8	COMM_STATUS	Bitstring(2)		0	E	D / RO	Indicate if communication from slave is good or not (each bit corresponds to a Modbus variable).
9	FLOCATOR1	DS-260				S / O/S	Information to locate float parameter
10	FVALUE1	Float		0		N	Value from requested address.
11	FLOCATOR2	DS-260				S / O/S	Information to locate float parameter
12	FVALUE2	Float		0		N	Value from requested address.
13	PLOCATOR1	DS-258				S / O/S	Information to locate percentage parameter

Index	Parameter	Data Type (length)	Valid Range/Options	Default Value	Units	Store / Mode	Description
14	PVALUE1	Float		0		N	Value from requested address.
15	PLOCATOR2	DS-258				S / O/S	Information to locate percentage parameter
16	PVALUE2	Float		0		N	Value from requested address.
17	ILOCATOR1	DS-260				S / O/S	Information to locate integer parameter
18	ILENGTH1	Integer8	1,2,4	2		S / O/S	Data length.
19	IVALUE1	Integer32		0		N	Value from requested address.
20	ILOCATOR2	DS-260				S / O/S	Information to locate integer parameter
21	ILENGTH2	Integer8	1,2,4	2		S / O/S	Data length.
22	IVALUE2	Integer32		0		N	Value from requested address.
23	BLOCATOR1	DS-260				S / O/S	Information to locate boolean parameter
24	BVALUE1	Boolean		TRUE		N	Value from requested addresses.
25	BLOCATOR2	DS-260				S / O/S	Information to locate boolean parameter
26	BVALUE2	Boolean		TRUE		N	Value from requested addresses.
27	BLOCATOR3	DS-260				S / O/S	Information to locate boolean parameter
28	BVALUE3	Boolean		TRUE		N	Value from requested addresses.
29	BLOCATOR4	DS-260				S / O/S	Information to locate boolean parameter
30	BVALUE4	Boolean		TRUE		N	Value from requested addresses.
31	BLOCATOR5	DS-260				S / O/S	Information to locate boolean parameter
32	BVALUE5	Boolean		TRUE		N	Value from requested addresses.
33	BLOCATOR6	DS-260				S / O/S	Information to locate boolean parameter
34	BVALUE6	Boolean		TRUE		N	Value from requested addresses.
35	BLOCATOR7	DS-260				S / O/S	Information to locate boolean parameter
36	BVALUE7	Boolean		TRUE		N	Value from requested addresses.
37	BLOCATOR8	DS-260				S / O/S	Information to locate boolean parameter
38	BVALUE8	Boolean		TRUE		N	Value from requested addresses.
39	UPDATE_EVT	DS-73			Na	D	This alert is generated by any change to the static data.
40	BLOCK_ALM	DS-72			Na	D	The block alarm is used for all configuration, hardware, connection failure or system problems in the block. The cause of the alert is entered in the subcode field. The first alert to become active will set the Active status in the Status attribute. As soon as the Unreported status is cleared by the alert reporting task, another block alert may be reported without clearing the Active status, if the subcode has changed.

Legend: E – Enumerated parameter; na – Adimensional parameter; RO – Read only; D – dynamic; N – non-volatile; S - static

Relation between the bits in COMM_STATUS and Modbus addresses

Bit	Variable
0	BAD COMM BMOD_ADD1
1	BAD COMM BMOD_ADD2
2	BAD COMM BMOD_ADD3
3	BAD COMM BMOD_ADD4
4	BAD COMM BMOD_ADD5
5	BAD COMM BMOD_ADD6
6	BAD COMM BMOD_ADD7
7	BAD COMM BMOD_ADD8
8	BAD COMM IMOD_ADD1
9	BAD COMM IMOD_ADD2
10	BAD COMM PMOD_ADD1
11	BAD COMM PMOD_ADD2
12	BAD COMM FMOD_ADD1
13	BAD COMM FMOD_ADD2

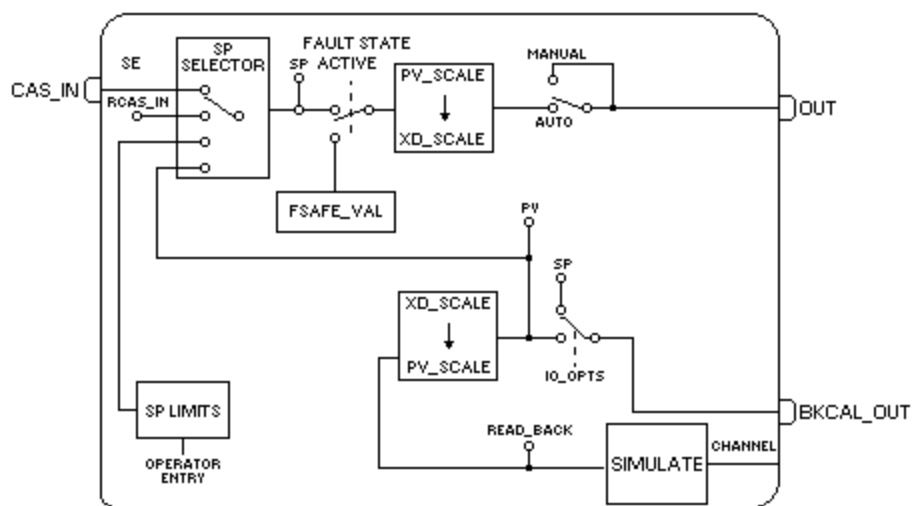
2.9 - OUTPUT FUNCTION BLOCKS

AO - Analog Output

Overview

The Analog Output Block is a function block used by devices that work as output elements in a control loop, like valves, actuators, positioners, etc. The AO block receives a signal from another function block and passes its results to an output transducer block through an internal channel reference.

Schematic



Description

The AO block is connected to the transducer block through the CHANNEL parameter that must match with the following parameter in the transducer block: TERMINAL_NUMBER parameter for the FI302

The CHANNEL parameter must be set to 1 (one) if the AO block is running in the FY302 or FP302, and no configuration is necessary in the transducer block to connect it to the AO block.

Treatment of Input Values

The SP value may be controlled automatically through a cascade or remote cascade control or manually by an operator. The PV_SCALE and XD_SCALE are used to do the scaling conversion of the SP.

Treatment of Output Values

The transducer scaling (XD_SCALE) is used to convert percent of span to the number used by the transducer. This allows portions of the SP span to cause full span movement of the output.

$$OUT = SP\% * (EU\_100\% - EU\_0\%) + EU\_0\% [XD\_SCALE]$$

The bit "Increase to Close" in IO_OPTS allows the output to be inverted relative to the span of the input value. For example, if the SP is 100. (PV_SCALE=0-100%; XD_SCALE = 3-15Psi):
If the "Increase to Close" bit in IO_OPTS is clear, SP converted to OUT_SCALE will be 15 psi.
Therefore the actuator type will be "air to open".

If the "Increase to Close" bit in IO_OPTS is true, SP converted to OUT_SCALE will be 3 psi.
Therefore the actuator type will be "air to close".

Simulate

The SIMULATE parameter is used for the diagnostics and checkout purposes. When it is active, the transducer value and status will be overridden by the simulate value and status. The SIMULATE can be disabled either by software in the SIMULATE parameter or hardware through the jumper.

The SIMULATE structure is composed by the following attributes:

Simulate Value and Status

Transducer Value and Status

Simulate Enable/Disable

The Transducer Value/Status attributes of SIMULATE parameter are always showing the value that the AO block receives from the corresponding transducer block.

There is a hardware jumper to disable the SIMULATE parameter. If this jumper is placed Off, then the simulation will be disabled. In this case, the user cannot change the ENABLE/DISABLE attribute. This jumper prevents simulation from accidentally being enabled during plant operations. When the jumper is placed ON, it will cause "Simulate Active" attribute in the BLOCK_ERR of Resource block to be true.

The simulate is active if the following conditions exist:

The simulate hardware jumper is not placed Off;

The SIMULATE.ENABLE/DISABLE parameter is "Active".

When simulation is active, the READBACK and PV parameters will be calculated based on the attribute Simulate Value/Status of the SIMULATE parameter. Otherwise it will be that one supplied by the transducer block in the Transducer Value/Status attribute of the SIMULATE parameter.

Readback parameter

If the hardware supports a readback value, such as valve position, then the value will be read by the transducer block and it will be provided to the corresponding AO block through the Transducer Value/Status attribute of the SIMULATE parameter. If not supported, the Transducer Value/Status attribute of the SIMULATE parameter is generated from AO.OUT by the transducer block.

The READBACK parameter has a copy of the Transducer Value/Status attribute of the SIMULATE parameter if the simulation is disabled, otherwise it is a copy of the Simulate Value/Status attribute of the SIMULATE parameter

The PV is the READBACK parameter converted to the PV_SCALE, therefore the PV can be simulated through the SIMULATE parameter.

In addition, the block admits safe action as described early in the fault state processing.

The AO block supports the mode-shedding feature as described early in the mode parameter.

BLOCK_ERR

The BLOCK_ERR of the AO block will reflect the following causes:

Block Configuration Error – the configuration error occurs when one or more of the following situations occur:

When the CHANNEL or SHED_OPT parameters have an invalid value;

When the XD_SCALE does not have a supported engineering unit and/or range for the respective the transducer block.

When the transducer block is in O/S mode.

When it is not compatible the CHANNEL parameter and HC configuration (DFI302).

Simulate Active – When the Simulate is active.

Local Override – When the block is in LO mode because the fault state is active.

Output Failure – I/O module failure (DFI302)

Out of Service – Occur when the block is in O/S mode.

Supported Modes

O/S, IMAN, LO, MAN, AUTO, CAS and RCAS.

Parameters

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
1	ST_REV	Unsigned16		0	None	S	
2	TAG_DESC	OctString(32)		Spaces	Na	S	
3	STRATEGY	Unsigned16		0	None	S	
4	ALERT_KEY	Unsigned8	1 to 255	0	None	S	
5	MODE_BLK	DS-69		O/S	Na	S	See Mode Parameter
6	BLOCK_ERR	Bitstring(2)			E	D	
7	PV	DS-65			PV	D / RO	Process analog value.
8	SP	DS-65	PV_SCALE +/- 10%		PV	N / Auto	The analog set point. Can be set manually, automatically through the interface device or another field device.
9	OUT	DS-65	XD_SCALE		OUT	N / Man	The output value result to the transducer block.
10	SIMULATE	DS-82	1: Disable ; 2: Active are the Enable/Disable options.	Disable		D	Allows the readback value to be manually supplied when simulate is enabled. In this case, the simulate value and status will be the PV value.
11	PV_SCALE	DS-68		0-100%	PV	S / Man	The high and low scale values to the SP parameter.
12	XD_SCALE	DS-68		0-100%	XD	S / Man	The high and low scale values, to transducer for a specified channel.

Index	Parameter	Data Type (length)	Valid Range/Options	Default Value	Units	Store / Mode	Description
13	GRANT_DENY	DS-70		0	Na	D	
14	IO_OPTS	Bitstring(2)	See Block Options	0	Na	S / O/S	See Block Options
15	STATUS_OPTS	Bitstring(2)	See Block Options	0	Na	S / O/S	See Block Options
16	READBACK	DS-65			XD	D / RO	Indicate the readback of the actual position of the transducer, in transducer units.
17	CAS_IN	DS-65				D	This parameter is the remote setpoint value, which must come from another Fieldbus block, or a DCS block through a defined link.
18	SP_RATE_DN	Float	Positive	+INF	PV/Sec	S	Ramp rate at which upward setpoint changes in PV units per second. It is disable if is zero or +INF. Rate limiting will apply in AUTO, CAS and RCAS modes.
19	SP_RATE_UP	Float	Positive	+INF	PV/Sec	S	Ramp rate at which downward setpoint changes in PV units per second. It is disable if is zero or +INF. Rate limiting will apply in AUTO, CAS and RCAS modes.
20	SP_HI_LIM	Float	PV_SCALE +/- 10%	100	PV	S	The setpoint high limit is the highest setpoint operator entry that can be used for the block.
21	SP_LO_LIM	Float	PV_SCALE +/- 10%	0	PV	S	The setpoint low limit is the lowest setpoint operator entry that can be used for the block.
22	CHANNEL	Unsigned16		0	None	S / O/S	The number of the logical hardware channel to the transducer that is connected to this I/O block.
23	FSTATE_TIME	Float	Positive	0	Sec	S	The time in seconds to ignore the existence of a new fault state condition. If the fault state condition does not persist for FSTATE_TIME seconds and while this time does not elapse, the block will execute in the last actual mode.
24	FSTATE_VAL	Float	PV_SCALE +/- 10%	0	PV	S	The preset analog SP value to use when fault occurs. This value be used if the I/O option fault state to value is selected.
25	BKCAL_OUT	DS-65			PV	D / RO	The value and status required by an upper block's BKCAL_IN so that the upper block may prevent reset windup and provide bumpless transfer to closed loop control.

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
26	RCAS_IN	DS-65			PV	D	Target setpoint and status provided by a supervisory Host to a analog control or output block.
27	SHED_OPT	Unsigned8	1: NormalShed, NormalReturn 2: NormalShed, NoReturn 3: ShedToAuto, NormalReturn 4: ShedToAuto, NoReturn 5: ShedToMan, NormalReturn 6: ShedToMan, NoReturn 7: ShedToRetained Target, NormalReturn 8: ShedToRetained Target, NoReturn	0		S	Defines action to be taken on remote control device timeout.
28	RCAS_OUT	DS-65			PV	D / RO	Block setpoint and status after ramping – provided to a supervisory Host for back calculation and to allow action to be taken under limiting conditions or mode change.
29	UPDATE_EVT	DS-73			Na	D	
30	BLOCK_ALM	DS-72			Na	D	

Legend: E – Enumerated parameter; na – Adimensional parameter; RO – Read only; D – dynamic;
N – non-volatile; S - static

If BEHAVIOR parameter is “Adapted”:

The default value of CHANNEL is the lowest available number.

The default value of SHED_OPT is NormalShed/NormalReturn.

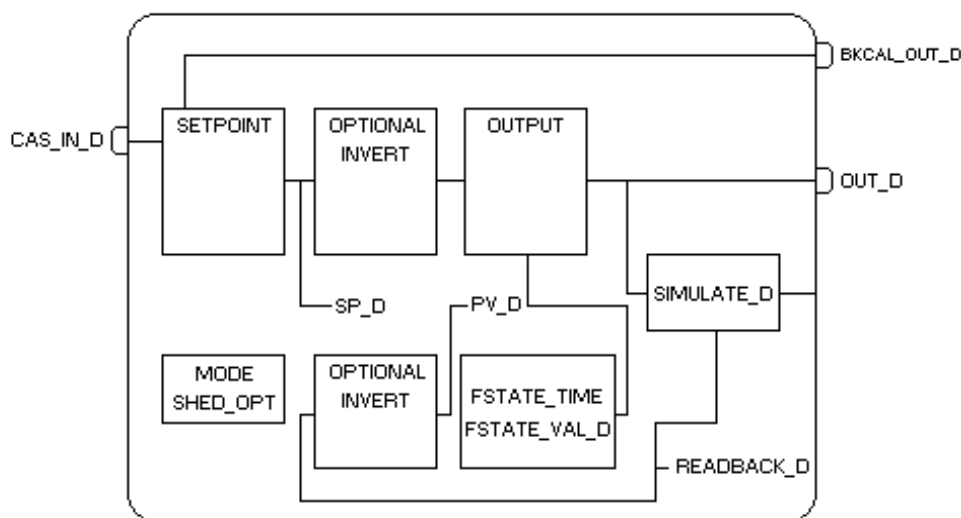
The required mode for writing is the actual mode, regardless the target mode : SP and OUT

DO - Discrete Output

Overview

The DO block converts the value in SP_D to something useful for the hardware found at the CHANNEL selection.

Schematic



Description

The Invert I/O option can be used to do a Boolean NOT function between the SP_D and the hardware.

The SP_D supports the full cascade sub-function. Cas mode must be used to transfer the output of another block to the SP_D of the DO.

There are additional I/O options which will cause the SP_D value to track the PV_D value when the block is in an actual mode of LO or Man.

If the hardware supports a readback value, it is used for READBACK_D, which, after accounting for the Invert I/O option, acts as the

PV_D for this block. If not supported, READBACK_D is generated from OUT_D. The OUT_D and READBACK_D parameters both use XD_STATE. The PV_D and SP_D use PV_STATE.

BLOCK_ERR

The BLOCK_ERR of the DO block will reflect the following causes:

Block Configuration Error – the configuration error occurs when one or more of the following situations occur:

When the CHANNEL or SHED_OPT parameters have an invalid value;

When it is not compatible the CHANNEL parameter and HC configuration (DFI302).

Simulate Active – When the Simulate is active.

Local Override – When the block is in LO mode because the fault state is active.

Output Failure – I/O module failure (DFI302)

Out of Service – Occur when the block is in O/S mode.

Supported Modes

O/S, LO, Iman, Man, Auto, Cas, and RCas. The Man mode can be used to force the output, in a PLC sense. It may be that Man mode is

not permitted, but it must be supported so that Man mode may be entered when leaving O/S. . The IMan mode is used to indicated that there is no path to the final element.

Alarm Types

Standard block alarm.

Mode Handling

Standard transition in and out of O/S.

Status Handling

The status values described in Output Parameter Formal Model of Part 1 apply, with the exception of the control sub-status values. The following options from STATUS_OPTS apply:
Propagate Fault Backward

Initialization

Standard.

Fault State Recovery

Standard.

Bad to Good Output Transition

Use the readback value, if available, or restore the NV value of OUT_D.

DO Output Block Access

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
1	ST_REV	Unsigned16		0	None	S	
2	TAG_DESC	OctString(32)		Spaces	Na	S	
3	STRATEGY	Unsigned16		0	None	S	
4	ALERT_KEY	Unsigned8	1 to 255	0	None	S	
5	MODE_BLK	DS-69		O/S	Na	S	See Mode Parameter
6	BLOCK_ERR	Bitstring(2)			E	D	
7	PV_D	DS-66			PV	D / RO	Either the primary discrete value for use in executing the function, or a process value associated with it. May also be calculated from the READBACK_D value of a DO block.
8	SP_D	DS-66	PV_STATE		PV	N / Auto	The discrete setpoint of this block.
9	OUT_D	DS-66	OUT_STATE		OUT	N / Man	The primary discrete value calculated as a result of executing the function.
10	SIMULATE_D	DS-83	1: Disable ; 2: Active are the Enable/Disable options.	Disable		D	Allows the transducer discrete input or output to the block to be manually supplied when simulate is enabled. When simulation is disabled, the simulate value and status track the actual value and status.
11	PV_STATE	Unsigned16		0	PV	S	Index to the text describing the states of a discrete PV.

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
12	XD_STATE	Unsigned16		0	XD	S	Index to the text describing the states of a discrete for the value obtained from the transducer.
13	GRANT_DENY	DS-70		0	Na	D	Options for controlling access of host computer and local control panels to operating, tuning and alarm parameters of the block.
14	IO_OPTS	Bitstring(2)	See Block Options	0	Na	S / O/S	See Block Options
15	STATUS_OPTS	Bitstring(2)	See Block Options	0	Na	S / O/S	See Block Options
16	READBACK_D	DS-66			XD	D / RO	This indicates the readback of the actual discrete valve or other actuator position, in the transducer state.
17	CAS_IN_D	DS-66			PV	D / RO	This parameter is the remote setpoint value of a discrete block, which must come from another Fieldbus block, or a DCS block through a defined link.
18	CHANNEL	Unsigned16		0	None	S / O/S	The number of the logical hardware channel to the transducer that is connected to this I/O block.
19	FSTATE_TIME	Float	Positive	0	Sec	S	The time in seconds to ignore the existence of a new fault state condition. If the fault state condition does not persist for FSTATE_TIME seconds and while this time does not elapse, the block will execute in the last actual mode.
20	FSTATE_VAL_D	Unsigned8		0	PV	S	The preset discrete SP_D value to use when fault occurs. This value will be used if the I/O option Fault State to value is selected.
21	BKCAL_OUT_D	DS-66			PV	D / RO	The output value and status provided to an upstream discrete block. This information is used to provide bumpless transfer to closed loop control.
22	RCAS_IN_D	DS-66			PV	D	Target setpoint and status provided by a supervisory Host to a discrete control or output block.

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
23	SHED_OPT	Unsigned8	1: NormalShed, NormalReturn 2: NormalShed, NoReturn 3: ShedToAuto, NormalReturn 4: ShedToAuto, NoReturn 5: ShedToMan, NormalReturn 6: ShedToMan, NoReturn 7: ShedToRetained Target, NormalReturn 8: ShedToRetained Target, NoReturn	0		S	Defines action to be taken on remote control device timeout.
24	RCAS_OUT_D	DS-66			PV	D / RO	Block setpoint and status provided to a supervisory Host for back calculation and to allow action to be taken under limiting conditions or mode change.
25	UPDATE_EVT	DS-73			Na	D	This alert is generated by any change to the static data.
26	BLOCK_ALM	DS-72			Na	D	The block alarm is used for all configuration, hardware, connection failure or system problems in the block. The cause of the alert is entered in the subcode field. The first alert to become active will set the Active status in the Status attribute. As soon as the Unreported status is cleared by the alert reporting task, another block alert may be reported without clearing the Active status, if the subcode has changed.

Legend: E – Enumerated parameter; na – Adimensional parameter; RO – Read only; D – dynamic; N – non-volatile; S - static

MAO - Multiple Analog Output

Description

The MAO block makes available to the I/O subsystem its eight input parameters IN_1 through IN_8.

This function block has the same fault state characteristics as the AO block. It includes option to hold the last value or go to a preset value when fault state active, individual preset values for each point, besides a delay time to go into the fault state.

The actual mode will be LO only due to the resource block, otherwise bad status in input parameter and configuration of MO_STATUS_OPTS will not affect the mode calculation. However the functionality of fault state will be done only for that input parameter. The parameter FSTATE_STATE shows which points are in fault state active.

BLOCK_ERR

The BLOCK_ERR of the MAO block will reflect the following causes:

Other – the number of MDI, MDO, MAI and MAO blocks or the device tag in FB700 is different from LC700.

Block Configuration Error – the configuration error occurs when the OCCURRENCE has an invalid value

Output failure – the CPU of LC700 stopped working.

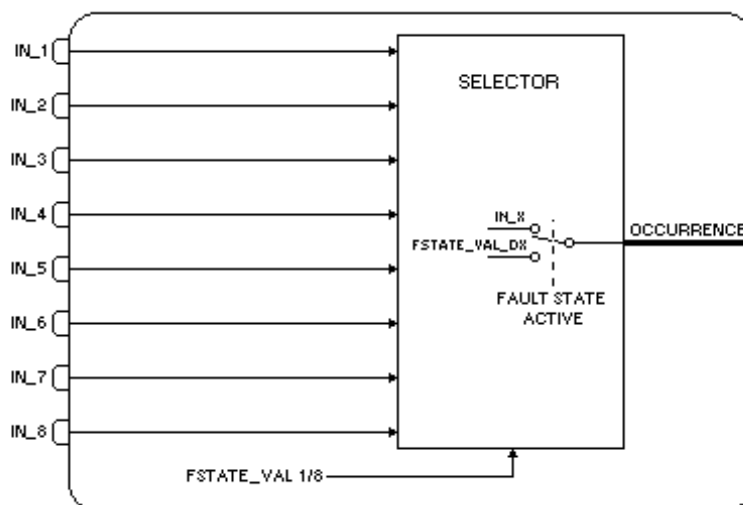
Power up – there is no CPU of LC700 in the rack or the hardware configuration of LC700 has an error

Out of Service – When the block is in O/S mode.

Supported Modes

O/S, LO and AUTO.

Schematic



Parameters

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
13	ST_REV	Unsigned16		0	None	S	
14	TAG_DESC	OctString(32)		Spaces	Na	S	
15	STRATEGY	Unsigned16		0	None	S	
4	ALERT_KEY	Unsigned8	1 to 255	0	None	S	
5	MODE_BLK	DS-69		O/S	Na	S	See Mode Parameter
6	BLOCK_ERR	Bitstring(2)			E	D	
7	OCCURRENCE / CHANNEL	Unsigned16		0	None	S / O/S	The number of the logical hardware OCCURRENCE/CHANNEL that is connected to this multiple I/O block. It defines the transducer to be used going to or from the physical world. It addresses a group of eight points.
8	IN_1	DS-65				D	Numbered input 1.
9	IN_2	DS-65				D	Numbered input 2.
10	IN_3	DS-65				D	Numbered input 3.
11	IN_4	DS-65				D	Numbered input 4.
12	IN_5	DS-65				D	Numbered input 5.
13	IN_6	DS-65				D	Numbered input 6.
14	IN_7	DS-65				D	Numbered input 7.
15	IN_8	DS-65				D	Numbered input 8.
16	MO_OPTS (different bit description in profile revision 1)	Bitstring(2)	See Block Options	0	Na	S / O/S	See Block Options
17	MO_STATUS_OPTS (not available in profile revision 1)	Bitstring(2)	See Block Options	0	Na	S / O/S	See Block Options
18	FSTATE_TIME	Float	Positive	0	Sec	S	The time in seconds to ignore the existence of a new fault state condition. If the fault state condition does not persist for FSTATE_TIME seconds and while this time does not elapse, the block will execute in the last actual mode.
19	FSTATE_VAL1	Float		0		S	The preset analog value to use when failure occurs in IN_1. Ignored if the "Fault state to value 1" in the MO_OPTS parameter is false.

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
20	FSTATE_VAL2	Float		0		S	The preset analog value to use when failure occurs in IN_2. Ignored if the "Fault state to value 2" in the MO_OPTS parameter is false.
21	FSTATE_VAL3	Float		0		S	The preset analog value to use when failure occurs in IN_3. Ignored if the "Fault state to value 3" in the MO_OPTS parameter is false.
22	FSTATE_VAL4	Float		0		S	The preset analog value to use when failure occurs in IN_4. Ignored if the "Fault state to value 4" in the MO_OPTS parameter is false.
23	FSTATE_VAL5	Float		0		S	The preset analog value to use when failure occurs in IN_5. Ignored if the "Fault state to value 5" in the MO_OPTS parameter is false.
24	FSTATE_VAL6	Float		0		S	The preset analog value to use when failure occurs in IN_6. Ignored if the "Fault state to value 6" in the MO_OPTS parameter is false.
25	FSTATE_VAL7	Float		0		S	The preset analog value to use when failure occurs in IN_7. Ignored if the "Fault state to value 7" in the MO_OPTS parameter is false.
26	FSTATE_VAL8	Float		0		S	The preset analog value to use when failure occurs in IN_8. Ignored if the "Fault state to value 8" in the MO_OPTS parameter is false.
27	FSTATE_STATE	Unsigned8			None	D / RO	It shows which points are in fault state active.
28	BLOCK_ALM	DS-72			Na	D	The block alarm is used for all configuration, hardware, connection failure or system problems in the block. The cause of the alert is entered in the subcode field. The first alert to become active will set the Active status in the Status attribute. As soon as the Unreported status is cleared by the alert reporting task, another block alert may be reported without clearing the Active status, if the subcode has changed.
29	UPDATE_EVT	DS-73			Na	D	This alert is generated by any change to the static data.

**Legend: E – Enumerated parameter; na – Adimensional parameter; RO – Read only;
D – dynamic; N – non-volatile; S - static**

If BEHAVIOR parameter is “Adapted”:

The default value of OCCURRENCE is the number of MAO blocks instantiated in the profile revision
0

Device type	Profile revision	Description
FB700	0	Block has OCCURRENCE parameter
DFI302	1	Block has CHANNEL parameter. MO_OPTS has a different bit description MO_STATUS_OPTS is not available in profile revision 1 There is a shift in the relative index for below parameters.

MDO - Multiple Discrete Output

Description

The MDO block makes available to the I/O subsystem its eight input parameters IN_D1 through IN_D8.

This function block has the same fault state characteristics as the DO block. It includes option to hold the last value or go to a preset value when fault state active, individual preset values for each point, besides a delay time to go into the fault state state.

The actual mode will be LO only due to the resource block, otherwise bad status in input parameter and configuration of MO_STATUS_OPTS will not affect the mode calculation. However the functionality of fault state will be done only for that input parameter.

The parameter FSTATE_STATE shows which points are in fault state active.

BLOCK_ERR

The BLOCK_ERR of the MDO block will reflect the following causes:

Other – the number of MDI, MDO, MAI and MAO blocks or the device tag in FB700 is different from LC700.

Block Configuration Error – the configuration error occurs when the OCCURRENCE / CHANNEL has an invalid value

Output failure – the CPU of LC700 stopped working.

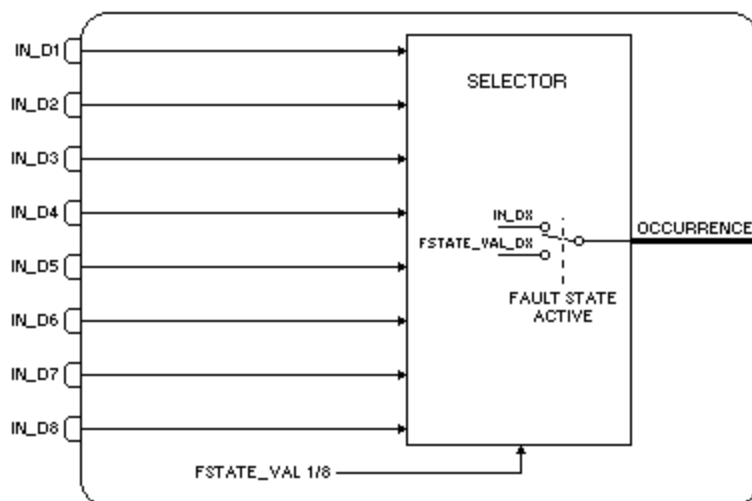
Power up – there is no CPU of LC700 in the rack or the hardware configuration of LC700 has an error

Out of Service – When the block is in O/S mode.

Supported Modes

O/S, LO and AUTO.

Schematic



Parameters

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
1	ST_REV	Unsigned16		0	None	S	
2	TAG_DESC	OctString(32)		Spaces	Na	S	
3	STRATEGY	Unsigned16		0	None	S	
4	ALERT_KEY	Unsigned8	1 to 255	0	None	S	
5	MODE_BLK	DS-69		O/S	Na	S	See Mode Parameter
6	BLOCK_ERR	Bitstring(2)			E	D	
7	OCCURRENCE / CHANNEL	Unsigned16		0	None	S / O/S	The number of the logical hardware OCCURRENCE/CHANNEL that is connected to this multiple I/O block. It defines the transducer to be used going to or from the physical world. It addresses a group of eight points.
8	IN_D1	DS-66				D	Numbered discrete input 1.
9	IN_D2	DS-66				D	Numbered discrete input 2.
10	IN_D3	DS-66				D	Numbered discrete input 3.
11	IN_D4	DS-66				D	Numbered discrete input 4.
12	IN_D5	DS-66				D	Numbered discrete input 5.
13	IN_D6	DS-66				D	Numbered discrete input 6.
14	IN_D7	DS-66				D	Numbered discrete input 7.
15	IN_D8	DS-66				D	Numbered discrete input 8.
16	MO_OPTS (different bit description in profile revision 1)	Bitstring(2)	See Block Options	0	Na	S / O/S	See Block Options
17	MO_STATUS_OPT S (not available in profile revision 1)	Bitstring(2)	See Block Options	0	Na	S / O/S	See Block Options
18	FSTATE_TIME	Float	Positive	0	Sec	S	The time in seconds to ignore the existence of a new fault state condition. If the fault state condition does not persist for FSTATE_TIME seconds and while this time does not elapse, the block will execute in the last actual mode.
19	FSTATE_VAL_D1	Unsigned8		0		S	The preset discrete value to use when failure occurs in IN_D1. Ignored if the "Fault state to value 1" in the MO_OPTS parameter is false.
20	FSTATE_VAL_D2	Unsigned8		0		S	The preset discrete value to use when failure occurs in IN_D2. Ignored if the "Fault state to value 2" in the MO_OPTS parameter is false.
21	FSTATE_VAL_D3	Unsigned8		0		S	The preset discrete value to use when failure occurs in IN_D3. Ignored if the "Fault state to value 3" in the MO_OPTS parameter is false.

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
22	FSTATE_VAL_D4	Unsigned8		0		S	The preset discrete value to use when failure occurs in IN_D4. Ignored if the "Fault state to value 4" in the MO_OPTS parameter is false.
23	FSTATE_VAL_D5	Unsigned8		0		S	The preset discrete value to use when failure occurs in IN_D5. Ignored if the "Fault state to value 5" in the MO_OPTS parameter is false.
24	FSTATE_VAL_D6	Unsigned8		0		S	The preset discrete value to use when failure occurs in IN_D6. Ignored if the "Fault state to value 6" in the MO_OPTS parameter is false.
25	FSTATE_VAL_D7	Unsigned8		0		S	The preset discrete value to use when failure occurs in IN_D7. Ignored if the "Fault state to value 7" in the MO_OPTS parameter is false.
26	FSTATE_VAL_D8	Unsigned8		0		S	The preset discrete value to use when failure occurs in IN_D8. Ignored if the "Fault state to value 8" in the MO_OPTS parameter is false.
27	FSTATE_STATE	Unsigned8			None	D / RO	It shows which points are in fault state active.
28	BLOCK_ALM	DS-72			Na	D	The block alarm is used for all configuration, hardware, connection failure or system problems in the block. The cause of the alert is entered in the subcode field. The first alert to become active will set the Active status in the Status attribute. As soon as the Unreported status is cleared by the alert reporting task, another block alert may be reported without clearing the Active status, if the subcode has changed.
29	UPDATE_EVT	DS-73			Na	D	This alert is generated by any change to the static data.

Legend: E – Enumerated parameter; na – Adimensional parameter; RO – Read only; D – dynamic; N – non-volatile; S - static

If BEHAVIOR parameter is "Adapted":

The default value of OCCURRENCE is the number of MDO blocks instantiated in the profile revision 0.

Device type	Profile revision	Description
FB700	0	Block has OCCURRENCE parameter
DFI302	1	Block has CHANNEL parameter. MO_OPTS has a different bit description MO_STATUS_OPTS is not available in profile revision 1 There is a shift in the relative index for below parameters.

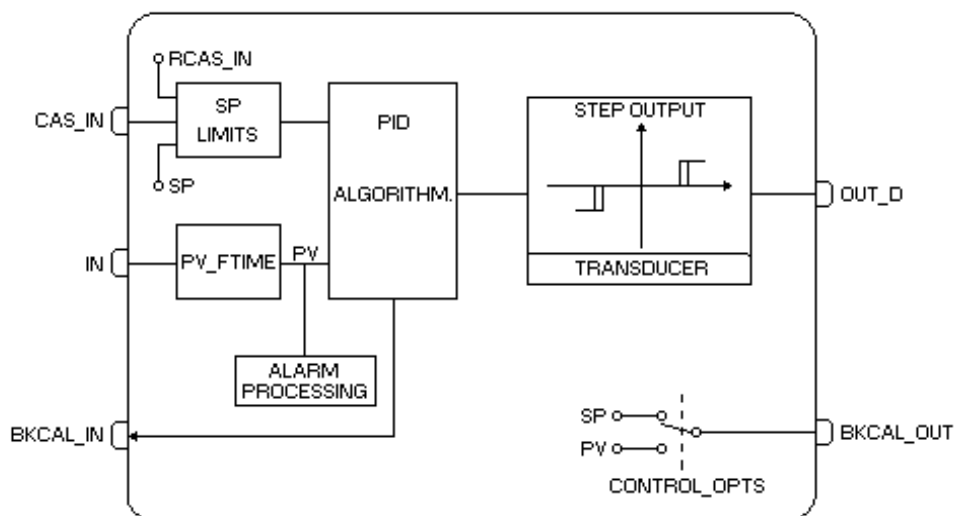
STEP – Step Output Pid

Overview

A Step Control Output block is used most commonly, when the final control element has an actuator driven by an electric motor. The final control element is positioned by rotating the motor clockwise or anticlockwise, which is accomplished by activating a discrete signal for each direction. A control valve, for example, needs a signal to open and another to close. If none of the signals is present, the valve stem would stay at the same position.

Fieldbus actuators and switchgears are the transducer blocks of this block.

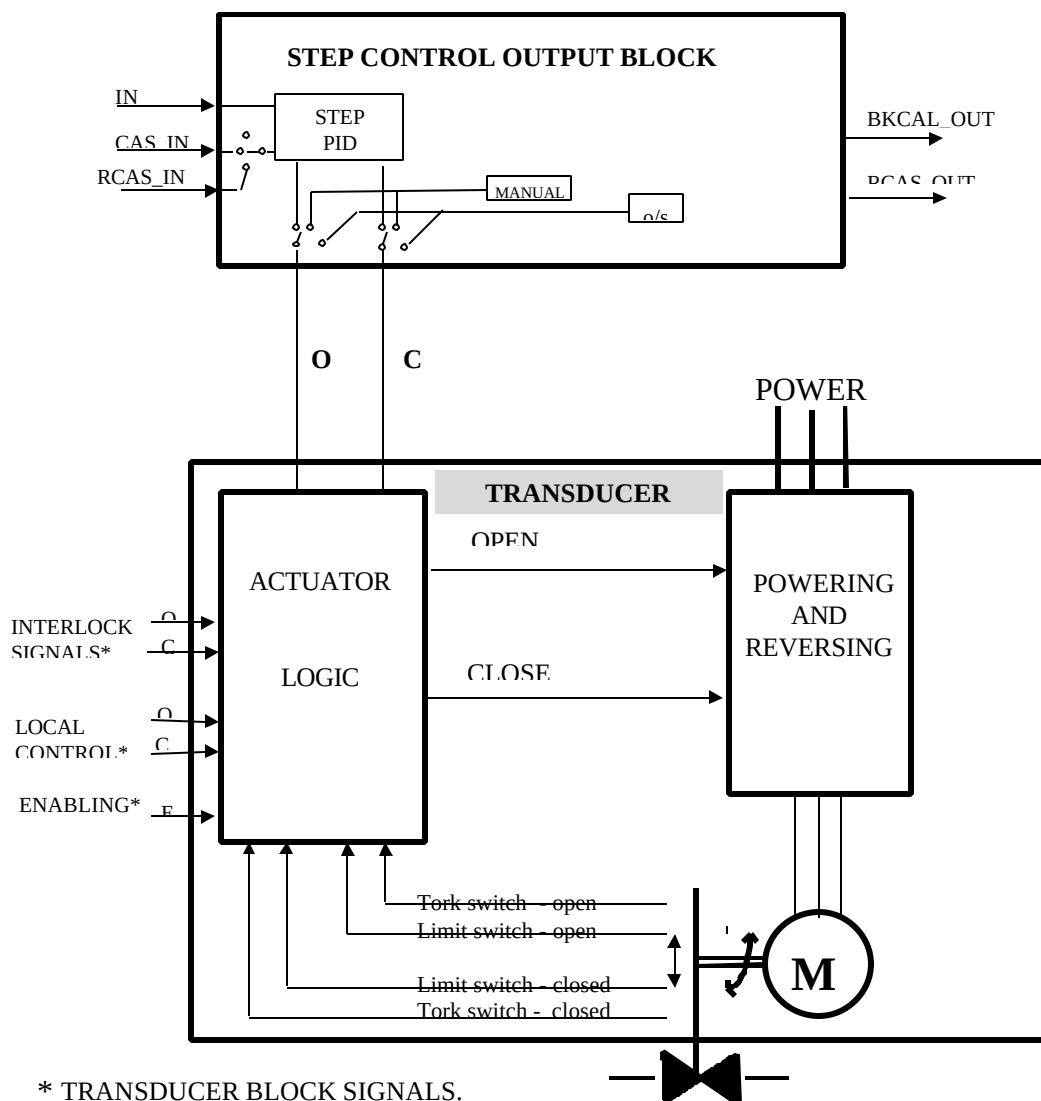
Schematic



Description

As shown on Fig.1, electric actuators require a switchgear module to power the electric motor and reverse it as demanded by the control loop. Most of the electric actuators require an interlock circuit to prevent the motor to overheat, or even burn, when the actuator reaches one of the travel limits or something blocks the movement in any direction, increasing the torque beyond an established limit. These actuators are normally equipped with torque switches and limit switches to provide this kind of protection.

The interlock circuit also establishes priorities for safety signals, manual commands, enabling signals and local commands, i.e., commands given from the actuator control panel or some sort of field mounted panel. Motor powering and reversing, interlocks and protection circuits can be combined in one Motor Control Center drawer or in a field device. In any case it will consist in a **Transducer**, that will convert the Fieldbus Function Block action into physical actions.



* TRANSDUCER BLOCK SIGNALS.
 Not defined in this specification.

Fig.1 Electric Actuator

Using a standard PID controller in cascade with a PI Step Controller.

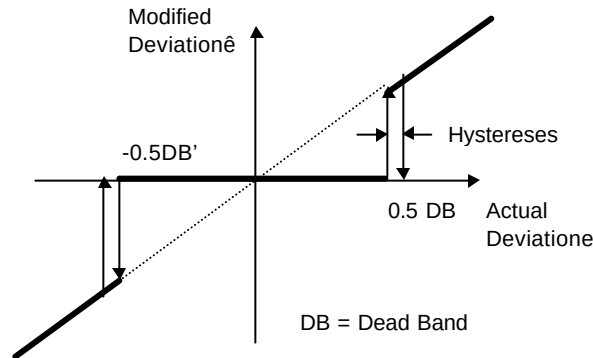


Fig. 4 - Modified deviation or gap deviation

The slave process variable is the position of the final control element, as shown in Fig.2.

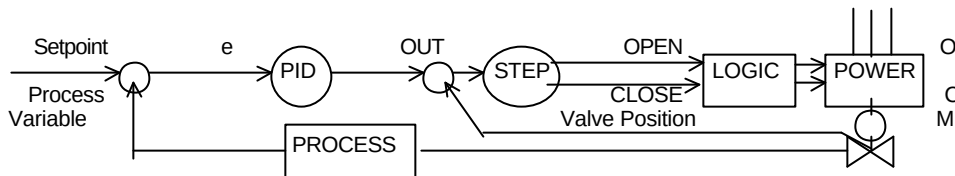


Fig. 2 - Step controller working as a positioner.

Controlling the process variable regardless of the valve position measurement.

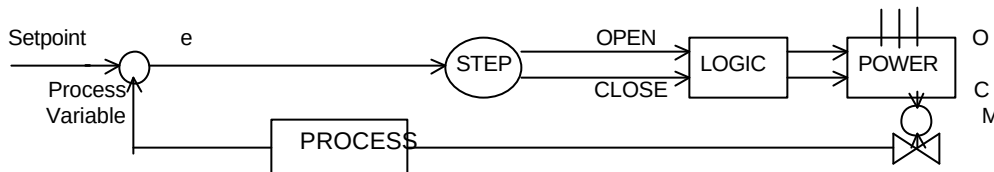


Fig. 3 - Step Controller

The actuator has a Travel Time, that is the time it takes to drive the final control element from one end limit to another. For a control valve, for example, it is the time required to drive it from closed to completely open.

The proportional action will actuate the final control element in the required direction during a time proportional to:

$$t_p = [\text{GAIN}] * (\hat{e} / 100) * [\text{TRAVEL\_TIME}] \dots (\text{s})$$

If the proportional action is not enough to turn $\hat{e} = 0$, the Integral Action will move the final control element at a speed of:

$$V = [\text{GAIN}] * \hat{e} / [\text{RESET}] \dots (\% / \text{s})$$

where the reset is the Integral time constant in seconds.

As most of the actuators work with constant and fixed speed, they can not give a speed larger than:

Maximum speed = $100\% / [\text{TRAVEL\_TIME}] \dots\dots (\% / \text{s})$

while the smaller speeds required by the Integral action are obtained by giving pulses of a specified duration [PULSE_DUR]. Each pulse will move the final control element a $\Delta x \%$ in the required direction.

$\Delta x \% = [\text{PULSE\_DUR}] * 100\% / [\text{TRAVEL\_TIME}] \dots\dots\dots (\%)$

The pulse frequency is given by:

$f = V / \Delta x \% \dots\dots\dots (\text{pulses} / \text{s})$

The Derivative or Rate action is given by:

$t_D = [\text{GAIN}] * (d\hat{e}/dt) * [\text{RATE}]$

where Rate is the derivative time constant in seconds and $d\hat{e}/dt$ can be calculated in several ways, including derivative gain, filtering, etc.

The PID Step Controller activates the OPEN or CLOSE signals according to the modified deviation, $\hat{e}$, the PID parameters and the other parameters in the following way:

The signal is activated during a time equivalent to:

$= t_P + t_D$

If the modified deviation is still different of zero, the Integral or Reset action will give pulses with a duration defined by [PULSE_DUR], with a frequency calculated by "f." t and f are dynamically modified by $\hat{e}$.

In order to avoid the Reset wind-up, the actuation time in one direction must be integrated and limited.

If the actuation time in one direction is larger than the [TRAVEL_TIME], there is no use in making the respective output signal to pulse, therefore it is recommendable to maintain it continuously activated.

The block provides a full PV and Deviation alarm support.

Supported Modes

Out-of-service, Manual, Auto, CAS and RCAS are supported.

In *Out-of-service* mode, the status of the output will be Out-of -service.

In *Manual* mode, the OUT_D can be set by the operator. The block stops output calculation.

In *Auto* mode the block operates normally.

In *CAS* (cascade), the Setpoint is supplied by another function block through the CAS_IN parameter.

In *RCAS* the block setpoint is set by a control application running on a computer, DCS or PLC.

Status Handling

The status of OUT_Di shall reflect the worst quality of the status of any connected input.

Initial Value Handling

The initial value of OUT_Di should be zero, ie, no action in both directions and the Integral action value should also go to zero.

Power Failure Recovery

After a power failure, OUT_Di should be Stop.

Parameters

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
1	ST_REV	Unsigned16		0	None	S	
2	TAG_DESC	OctString(32)		Spaces	Na	S	
3	STRATEGY	Unsigned16		0	None	S	
4	ALERT_KEY	Unsigned8	1 to 255	0	None	S	
5	MODE_BLK	DS-69		O/S	Na	S	See Mode Parameter
6	BLOCK_ERR	Bitstring(2)			E	D / RO	
7	PV	DS-65			PV	D / RO	Process analog value. This is the IN value after pass over the PV filter.
8	SP	DS-65	PV_SCALE +/- 10%		PV	N / Auto	The analog set point. Can be set manually, automatically through the interface device or another field device.
9	OUT_D	DS-66				N / Man	The output value result of the Step Output PID calculation.
10	PV_SCALE	DS-68		0-100%	PV	S / Man	The high and low scale values to the PV and SP parameter.
11	XD_STATE	Unsigned16		0	XD	S	Index to the text describing the states of a discrete for the value obtained from the transducer.
12	GRANT_DENY	DS-70		0	na	D	Options for controlling access of host computer and local control panels to operating, tuning and alarm parameters of the block.
13	CONTROL_OPTS	Bitstring(2)	See Block Options	0	na	S / O/S	See Block Options
14	STATUS_OPTS	Bitstring(2)	See Block Options	0	Na	S / O/S	See Block Options
15	IN	DS-65			PV	D	The primary input value of the block, or PV value.
16	PV_FTIME	Float	Non-Negative	0	Sec	S	Time constant of a single exponential filter for the PV, in seconds.
17	JOG_TIME	Float	Positive	0	Sec	S	Duration of OUT_D in the active state when commanded by the operator to jog open or jog closed.
18	CAS_IN	DS-65				D	This parameter is the remote setpoint value, which must come from another Fieldbus block, or a DCS block through a defined link.

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
19	SP_RATE_DN	Float	Positive	+INF	PV/Sec	S	Ramp rate at which upward setpoint changes in PV units per second. It is disable if is zero or +INF. Rate limiting will apply only in AUTO mode.
20	SP_RATE_UP	Float	Positive	+INF	PV/Sec	S	Ramp rate at which downward setpoint changes in PV units per second. It is disable if is zero or +INF. Rate limiting will apply only in AUTO mode.
21	SP_HI_LIM	Float	PV_SCALE +/- 10%	100	PV	S	The setpoint high limit is the highest setpoint operator entry that can be used for the block.
22	SP_LO_LIM	Float	PV_SCALE +/- 10%	0	PV	S	The setpoint low limit is the lowest setpoint operator entry that can be used for the block.
23	GAIN	Float		0	None	S	Proportional term of the PID. It is the Kp value.
24	RESET	Float	Positive	+INF	sec	S	Integral term of the PID. It is the Tr value.
25	BAL_TIME	Float	Positive	0	sec	S	This specifies the time for the internal working value of bias or ratio to return to the operator set bias or ratio, in seconds. In the PID block, it may be used to specify the time constant at which the integral term will move to obtain balance when the output is limited and the mode is Auto, Cas, or RCas.
26	RATE	Float	Positive	0	sec	S	Derivative term of the PID. It is the Td value.
27	IO_OPTS	Bitstring(2)	See Block Options	0	na	S / O/S	See Block Options
28	CHANNEL	Unsigned16		0	None	S / O/S	The number of the logical hardware channel to the transducer that is connected to this I/O block. In the DFI302, this parameter is selecting two discrete outputs. The first one is the CLOSE output and the next point in the same group will be the OPEN output. The CHANNEL parameter will be addressing the CLOSE output, despite of it is allocating the OPEN output too.

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
29	FSTATE_TIME	Float	Positive	0	Sec	S	The time in seconds from detection of fault of the output block remote setpoint to the output action of the block output if the condition still exists.
30	FSTATE_VAL_D	Unsigned8		0		S	The preset discrete SP_D value to use when fault occurs. This value will be used if the I/O option Fault State to value is selected.
31	BKCAL_OUT	DS-65			PV	D / RO	The value and status required by an upper block's BKCAL_IN so that the upper block may prevent reset windup and provide bumpless transfer to closed loop control.
32	RCAS_IN	DS-65			PV	D	Target setpoint and status provided by a supervisory Host to a analog control or output block.
33	SHED_OPT	Unsigned8	1: NormalShed, NormalReturn 2: NormalShed, NoReturn 3: ShedToAuto, NormalReturn 4: ShedToAuto, NoReturn 5: ShedToMan, NormalReturn 6: ShedToMan, NoReturn 7: ShedToRetained Target, NormalReturn 8: ShedToRetained Target, NoReturn	0		S	Defines action to be taken on remote control device timeout.
34	RCAS_OUT	DS-65			PV	D / RO	Block setpoint and status after ramping - provided to a supervisory Host for back calculation and to allow action to be taken under limiting conditions or mode change.
35	TRAVEL_TIME	Float	Positive	60	Sec	S / Man	The time required by the actuator to drive the final control element from one end position to another, in seconds.

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
36	PULSE_DUR	Float	Positive	1	Sec	S / Man	It is the width, in seconds, of the pulses given due to the integral action.
37	DEAD_BAND	Float	Non-negative	0	%	S / Man	It is the interval where changes in the Input will not change the output
38	HYSTERESIS	Float	Non-negative	0	%	S / Man	Difference between the switching points.
39	UPDATE_EVT	DS-73			Na	D	This alert is generated by any change to the static data.
40	BLOCK_ALM	DS-72			Na	D	The block alarm is used for all configuration, hardware, connection failure or system problems in the block. The cause of the alert is entered in the subcode field. The first alert to become active will set the Active status in the Status attribute. As soon as the Unreported status is cleared by the alert reporting task, another block alert may be reported without clearing the Active status, if the subcode has changed.
41	ALARM_SUM	DS-74	See Block Options		Na	S	The current alert status, unacknowledged states, unreported states, and disabled states of the alarms associated with the function block.
42	ACK_OPTION	Bitstring(2)	0: Auto ACK Disable 1: Auto ACK Enable	0	Na	S	Selection of whether alarms associated with the block will be automatically acknowledged
43	ALARM_HYS	Float	0 to 50 %	0.5%	%	S	Alarm hysteresis parameter. In order to clear the alarm the amount the PV must return within the alarm limit plus hysteresis.
44	HI_HI_PRI	Unsigned8	0 to 15	0		S	Priority of the high high alarm.
45	HI_HI_LIM	Float	OUT_SCALE, +INF	+INF	PV	S	The setting for high high alarm in engineering units.
46	HI_PRI	Unsigned8	0 to 15	0		S	Priority of the high alarm.
47	HI_LIM	Float	OUT_SCALE, +INF	+INF	PV	S	The setting for high alarm in engineering units.
48	LO_PRI	Unsigned8	0 to 15	0		S	Priority of the low alarm.
49	LO_LIM	Float	OUT_SCALE, -INF	-INF	PV	S	The setting for low alarm in engineering units.
50	LO_LO_PRI	Unsigned8	0 to 15	0		S	Priority of the low low alarm.
51	LO_LO_LIM	Float	OUT_SCALE, -INF	-INF	PV	S	The setting for low low alarm in engineering units.

Index	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
52	DV_HI_PRI	Unsigned8	0 to 15	0		S	Priority of the deviation high alarm.
53	DV_HI_LIM	Float	0 to PV span, +INF	+INF	PV	S	The setting for deviation high alarm in engineering units.
54	DV_LO_PRI	Unsigned8	0 to 15	0		S	Priority of the deviation low alarm.
55	DV_LO_LIM	Float	-INF, -PV span to 0	-INF	PV	S	The setting for deviation low alarm in engineering units.
56	HI_HI_ALM	DS-71			PV	D	The status for high high alarm and its associated time stamp.
57	HI_ALM	DS-71			PV	D	The status for high alarm and its associated time stamp.
58	LO_ALM	DS-71			PV	D	The status for low alarm and its associated time stamp.
59	LO_LO_ALM	DS-71			PV	D	The status for low low alarm and its associated time stamp.
60	DV_HI_ALM	DS-71			PV	D	The status for deviation high alarm and its associated time stamp.
61	DV_LO_ALM	DS-71			PV	D	The status for deviation low alarm and its associated time stamp.

Legend: E – Enumerated parameter; na – Adimensional parameter; RO – Read only; D – dynamic; N – non-volatile; S - static

Block Options

Resource Block Bit Strings

HARD_TYPES

Hardware types supported

Bit	Meaning
0	Scalar input (LSB)
1	Scalar output
2	Discrete input
3	Discrete output
4	
5	
6	
7	
8	
9	
10	
11	
12	
13	
14	
15	

CYCLE_TYPE and CYCLE_SEL

Types of cycle supported

Bit	Meaning
0	Scheduled (LSB)
1	Completion of block execution
2	Manufacturer specific
3	
4	
5	
6	
7	
8	
9	
10	
11	
12	
13	
14	
15	

FEATURES and FEATURE_SEL
Things that this resource supports

Bit	Meaning
0	Unicode strings
1	Reports supported
2	Fault State supported
3	Soft Write lock supported
4	Hard Write lock supported
5	Output readback supported
6	Direct write to output hardware
7	Change of bypass in an automatic mode
8	MVC supported
9	
10	
11	
12	
13	
14	
15	

Order of Resource Block Alerts
For ALARM_SUM and ACK_OPTION

Bit	Meaning
0	Writes have been enabled
1	
2	
3	
4	
5	
6	
7	Block alarm
8	
9	
10	
11	
12	
13	
14	
15	

Unicode strings

This feature is not supported.

Reports supported

It is necessary to set this feature in order to enable alert reporting in the resource.

Fault State supported

If this feature is selected in FEATURE_SEL, setting the SET_FSTATE parameter will force all output function blocks (AO and DO) in the resource to go to fault state.

Individual output function block will go to Fault State due to a loss of communication to CAS_IN or IFS status in CAS_IN, regardless the selection of this feature.

Soft Write lock supported

It is necessary to select this feature to set the WRITE_LOCK parameter.

Hard Write lock supported

This feature is not supported.

Output readback supported

Only the FY302 supports output readback and this feature is used regardless the selection in FEATURE_SEL.

Direct write to output hardware

This feature is not supported.

Change of BYPASS in an automatic mode

If this feature is selected in FEATURE_SEL, it is allowed to write in BYPASS in an automatic mode, otherwise only in Man or O/S modes.

MVC supported

The selection of this feature allows to optimize communication performance by transferring a grouped data as a single variable list in either publisher/subscriber transactions for function block links, or report distribution to a host device.

Function Block Options

IO_OPTS

Bit	Meaning	AI	DI	AO	DO	STEP
0	Invert (LSB)		X		X	
1	SP-PV Track in Man			X	X	
2	Reserved					
3	SP-PV Track in LO			X	X	
4	SP Track retained target			X	X	
5	Increase to close			X		
6	FAULT STATE to value			X	X	X
7	Use FAULT STATE value on restart			X	X	X
8	Target to Man if FAULT STATE actived			X	X	X
9	Use PV for BKCAL_OUT			X	X	
10	Low cutoff	X				
11	Reserved					
12	Reserved					
13	Reserved					
14	Reserved					
15	Reserved					

Invert

Indicates whether the discrete input value should be logically inverted before it is stored in the process variable.

SP-PV Track in Man

Permits the setpoint to track the process variable when the target mode of the block is Man.

SP-PV Track in LO

Permits the setpoint to track the process variable when the actual mode of the block is LO. IMan is not possible in an I/O block.

SP TRACKS RCAS OR CAS if LO OR MAN

Permits the setpoint to track the Rcas or Cas parameter based on the retained target mode when the actual mode of the block is LO or Man.

Increase to close

Indicates whether the output value should be inverted before it is communicated to the I/O channel.

FAULT STATE to value

The output action to take when failure occurs. (0: freeze, 1: go to preset value)

Use FAULT STATE value on restart

Use the value of FSTATE_VAL if the device is restarted, otherwise use the non-volatile value. This does not act like Fault State, just uses the value.

Target to Man if FAULT STATE activated

Set the target mode to Man, thus losing the original target, if Fault State is activated. This latches an output block into the manual mode.

Use PV for BKCAL_OUT

The BKCAL_OUT value is normally the working SP. This option changes it to the PV.

Low cutoff

The AI low cutoff algorithm is enabled.

CONTROL_OPTS

Bit	Meaning	PID	EPID	APID	CHAR	STEP
0	Bypass Enable (LSB)	X	X	X	X	
1	SP-PV Track in Man	X	X	X		X
2	SP-PV Track in Rout	X	X	X		
3	SP-PV Track in LO or IMan	X	X	X		X
4	SP Track Retained Target	X	X	X		X
5	Direct Acting	X	X	X		X
6	Reserved					
7	Track Enable	X	X	X		
8	Track in Manual	X	X	X		
9	Use PV for BKCAL_OUT	X	X	X		X
10	ACT on IR					
11	Use BKCAL_OUT with IN_1					
12	Obey SP limits if Cas or Rcas	X	X	X		X
13	No OUT limits in Manual	X	X	X		
14	Reserved					
15	Reserved					

Bypass Enable

This parameter, if true, allows BYPASS to be set. Some control algorithm applications cannot provide closed loop control if bypassed.

SP-PV Track in Man

Permits the setpoint to track the process variable when the target mode of the block is Man.

SP-PV Track in Rout

Permits the setpoint to track the process variable when the actual mode of the block is ROut.

SP-PV Track in LO or IMan

Permits the setpoint to track the process variable when the actual mode of the block is LO or IMan.

SP Track retained target

Permits the setpoint to track the Rcas or Cas parameter based on the retained target mode when the actual mode of the block is IMAN, LO, Man or ROut. When SP-PV Track options are enable, then SP track retained target will have precedence in the selection of the value to track when the actual mode is MAN, IMAN, ROUT and LO.

Direct Acting

Defines the relationship between a change in PV and corresponding change in output. When Direct is selected, an increase in PV results in an increase in the output.

Track Enable

This enables the external tracking function. If true, the value in TRK_VAL will replace the value of OUT if TRK_IN_D becomes true and the target mode is not Man.

Track in Manual

This enables TRK_VAL to replace the value of OUT when the target mode is Man and TRK_IN_D is true. The actual mode will then be LO.

Use PV for BKCAL_OUT

The BKCAL_OUT and RCAS_OUT values are normally the working SP. If this options is enable, then the PV value will be used after the CASCADE is closed.

Act on IR

Normally, initialization only acts on PID blocks. Other blocks just pass it on. If this option is true, the BIAS term of BG or PD blocks, or the ratio SP, may be adjusted to initialize the output.

Use percent for IN_1

Normally, IN_1 has the same scaling as OUT in a Ratio block. If this option is set, the contents of IN_1 should be converted to the range of OUT_SCALE. This allows a pacing controller to drive a number of Ratio blocks. May also be useful for the Bias/Gain block.

Obey SP limits if Cas or Rcas

Normally the setpoint will not be restricted to the setpoint limits except when entered by a human interface device. However, if this option is selected, the setpoint will be restricted to the setpoint absolute limits in the Cas and Rcas modes.

No OUT limits in Manual

Do not apply OUT_HI_LIM or OUT_LO_LIM when target and actual mode are Man.. Trust the operator to do the right thing.

STATUS_OPTS

Bit	Meaning	AI	DI	PUL	PI D	EPID	APID	SPL T	AAL M	ISEL	SPG	TIME	LLAG	DEN S	FFE T	AO	DO	STEP
0	IFS if BAD IN (LSB)				X	X	X											
1	IFS if BAD CAS_IN				X	X	X	X										
2	Use Uncertain as Good				X	X	X	X	X	X	X	X	X	X	X			X
3	Propagate Fault Forward	X	X	X														
4	Propagate Fault Backward															X	X	X
5	Target to Manual if BAD IN				X	X	X											X
6	Uncertain if Limited	X		X														
7	BAD if Limited	X		X														
8	Uncertain if Man mode	X	X	X														
9	Target to next permitted mode if BAD CAS_IN				X	X	X											X
10	Reserved																	
11	Reserved																	
12	Reserved																	
13	Reserved																	
14	Reserved																	
15	Reserved																	

IFS if BAD IN

Set Initiate Fault State status in the OUT parameter if the status of the IN parameter is BAD.

IFS if BAD CAS_IN

Set Initiate Fault State status in the OUT parameter if the status of the CAS_IN parameter is BAD.

Use Uncertain as Good

If the status of the IN parameter is Uncertain, treat it as Good. Otherwise, treat it as BAD.

Propagate Fault Forward

If the status from the sensor is Bad, Device failure or Bad, Sensor failure, propagate it to OUT without generating an alarm. The use of these sub-status in OUT is determined by this option. Through this option, the user may determine whether alarming (sending of an alert) will be done by the block or propagated downstream for alarming.

Propagate Fault Backward

If the status from the actuator is Bad, Device failure or Fault State Active or Local Override is active, propagate this as Bad, Device Failure or Good Cascade, Fault State Active or Local Override to BKCAL_OUT respectively without generating an alarm. The use of these sub-status in BKCAL_OUT is determined by this option. Through this option, the user may determine whether alarming (sending of an alert) will be done by the block or propagated upstream for alarming.

Target to Manual if BAD IN

Set the target mode to Man if the status of the IN parameter is BAD. This latches a PID block into the Man state if the input ever goes bad.

Uncertain if Limited

Set the output status of an input or calculation block to uncertain if the measured or calculated value is limited.

BAD if Limited

Set the output status to Bad if the sensor is at a high or low limit.

Uncertain if Man Mode

Set the output status of an input or calculation block to uncertain if the actual mode of the block is Man.

ALARM_SUM and ACK_OPTION

(Valid for all blocks, except for Resource Block)

Bit	Meaning
0	Discrete alarm
1	High High alarm
2	High alarm
3	Low Low alarm
4	Low alarm
5	Deviation High alarm
6	Deviation Low alarm
7	Block alarm
8	
9	
10	
11	
12	
13	
14	
15	

APID AND EPID FUNCTION BLOCKS OPTIONS

PID_OPTS

Bit	Meaning
0	IFS if Bad TRK_IN_D
1	IFS if Bad TRK_VAL
2	Man if Bad TRK_IN_D
3	Man if Bad TRK_VAL
4	Target to Manual if BAD TRK_IN_D

5	Target to Manual if BAD TRK_VAL
6	Reserved
7	Reserved
8	Reserved
9	Reserved
10	Reserved
11	Reserved
12	Reserved
13	Reserved
14	Reserved
15	Reserved

INTEGRATOR FUNCTION BLOCK OPTIONS

INTEG_OPTS

Bit	Meaning
0	Input 1 accumulate
1	Input 2 accumulate
2	Flow forward
3	Flow reverse
4	Use Uncertain
5	Use Bad
6	Carry
7	Reserved
8	Reserved
9	Reserved
10	Reserved
11	Reserved
12	Reserved
13	Reserved
14	Reserved
15	Reserved

TIMER FUNCTION BLOCK OPTIONS

INVERT_OPTS

Bit	Meaning
0	Invert IN_D1
1	Invert IN_D2
2	Invert IN_D3
3	Invert IN_D4
4	Invert OUT_D
5	Invert OUT_ALM
6	Reserved
7	Reserved
8	Reserved
9	Reserved
10	Reserved

Bit	Meaning
11	Reserved
12	Reserved
13	Reserved
14	Reserved
15	Reserved

ARITHMETIC FUNCTION BLOCK OPTIONS

INPUT_OPTS

Bit	Meaning
0	IN Use uncertain
1	IN_LO Use uncertain
2	IN_1 Use uncertain
3	IN_1 Use bad
4	IN_2 Use uncertain
5	IN_2 Use bad
6	IN_3 Use uncertain
7	IN_3 Use bad
8	Reserved
9	Reserved
10	Reserved
11	Reserved
12	Reserved
13	Reserved
14	Reserved
15	Reserved

OUTPUT SIGNAL SELECTOR AND DYNAMIC LIMITER FUNCTION BLOCK OPTIONS

OSDL_OPTS

Bit	Meaning
0	IFS if BAD IN
1	IFS if BAD CAS_IN
2	Use Uncertain as Good
3	Reserved
4	Reserved
5	Reserved
6	Reserved
7	Reserved
8	Reserved
9	Reserved
10	Reserved
11	IFS if BAD IN_1
12	Keep last value if not select
13	Reserved
14	Reserved
15	Reserved

MULTIPLE OUTPUT FUNCTION BLOCKS OPTIONS

MO_STATUS_OPTS

Bit	Meaning
0	IFS if BAD IN_1
1	IFS if BAD IN_2
2	IFS if BAD IN_3
3	IFS if BAD IN_4
4	IFS if BAD IN_5
5	IFS if BAD IN_6
6	IFS if BAD IN_7
7	IFS if BAD IN_8
8	Reserved
9	Reserved
10	Reserved

Bit	Meaning
11	Reserved
12	Reserved
13	Reserved
14	Reserved
15	Reserved

MO_OPTS (Profile Rev. 0 – FB700)

Bit	Meaning
0	Fault state to value 1
1	Use fault state value on restart 1
2	Fault state to value 2
3	Use fault state value on restart 2
4	Fault state to value 3
5	Use fault state value on restart 3
6	Fault state to value 4
7	Use fault state value on restart 4
8	Fault state to value 5
9	Use fault state value on restart 5
10	Fault state to value 6
11	Use fault state value on restart 6
12	Fault state to value 7
13	Use fault state value on restart 7
14	Fault state to value 8
15	Use fault state value on restart 8

MO_OPTS (Profile Rev. 1 – DFI302)

Bit	Meaning
0	Fault state to value 1
1	Fault state to value 2
2	Fault state to value 3
3	Fault state to value 4
4	Fault state to value 5
5	Fault state to value 6

6	Fault state to value 7
7	Fault state to value 8
8	Use fault state value on restart 1
9	Use fault state value on restart 2
10	Use fault state value on restart 3
11	Use fault state value on restart 4
12	Use fault state value on restart 5
13	Use fault state value on restart 6
14	Use fault state value on restart 7
15	Use fault state value on restart 8

HARDWARE CONFIGURATION BLOCK OPTIONS

MODULE_STATUS_RO_3

Bit	Meaning
0	Status of module in rack 0 slot 0
1	Status of module in rack 0 slot 1
2	Status of module in rack 0 slot 2
3	Status of module in rack 0 slot 3
4	Status of module in rack 1 slot 0
5	Status of module in rack 1 slot 1
6	Status of module in rack 1 slot 2
7	Status of module in rack 1 slot 3

Bit	Meaning
0	Status of module in rack 2 slot 0
1	Status of module in rack 2 slot 1
2	Status of module in rack 2 slot 2
3	Status of module in rack 2 slot 3
4	Status of module in rack 3 slot 0
5	Status of module in rack 3 slot 1
6	Status of module in rack 3 slot 2
7	Status of module in rack 3 slot 3

MODULE_STATUS_R4_7

Bit	Meaning
0	Status of module in rack 4 slot 0
1	Status of module in rack 4 slot 1
2	Status of module in rack 4 slot 2
3	Status of module in rack 4 slot 3
4	Status of module in rack 5 slot 0
5	Status of module in rack 5 slot 1

6	Status of module in rack 5 slot 2
7	Status of module in rack 5 slot 3

Bit	Meaning
0	Status of module in rack 6 slot 0
1	Status of module in rack 6 slot 1
2	Status of module in rack 6 slot 2
3	Status of module in rack 6 slot 3
4	Status of module in rack 7 slot 0
5	Status of module in rack 7 slot 1
6	Status of module in rack 7 slot 2
7	Status of module in rack 7 slot 3

MODULE_STATUS_R8_11

Bit	Meaning
0	Status of module in rack 8 slot 0
1	Status of module in rack 8 slot 1
2	Status of module in rack 8 slot 2
3	Status of module in rack 8 slot 3
4	Status of module in rack 9 slot 0
5	Status of module in rack 9 slot 1
6	Status of module in rack 9 slot 2
7	Status of module in rack 9 slot 3

Bit	Meaning
0	Status of module in rack 10 slot 0
1	Status of module in rack 10 slot 1
2	Status of module in rack 10 slot 2
3	Status of module in rack 10 slot 3
4	Status of module in rack 11 slot 0
5	Status of module in rack 11 slot 1
6	Status of module in rack 11 slot 2
7	Status of module in rack 11 slot 3

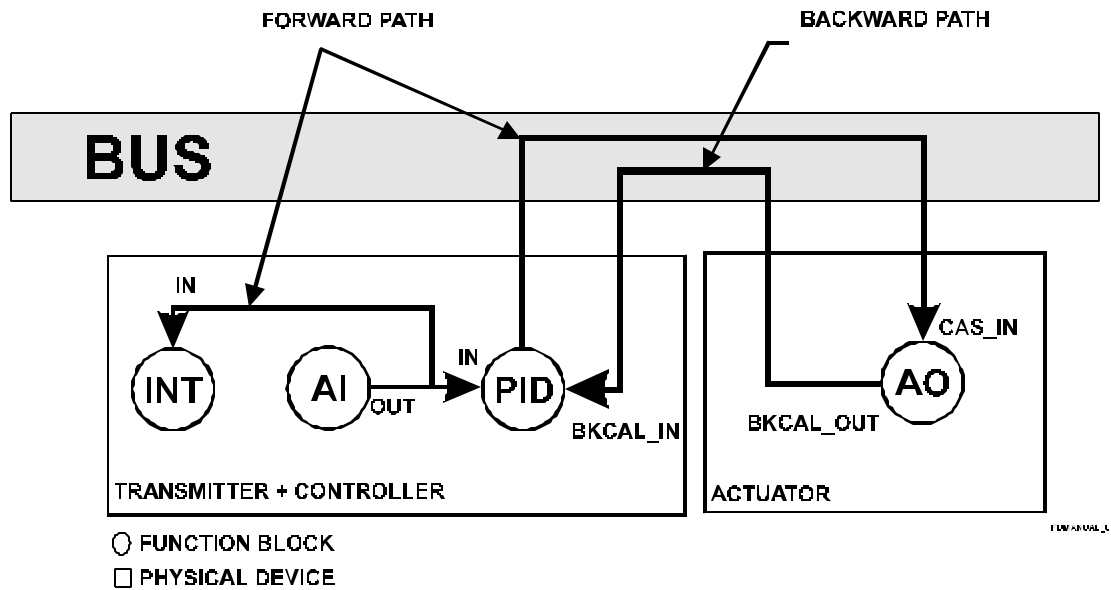
MODULE_STATUS_R12_14

Bit	Meaning
0	Status of module in rack 12 slot 0
1	Status of module in rack 12 slot 1
2	Status of module in rack 12 slot 2
3	Status of module in rack 12 slot 3
4	Status of module in rack 13 slot 0
5	Status of module in rack 13 slot 1
6	Status of module in rack 13 slot 2
7	Status of module in rack 13 slot 3

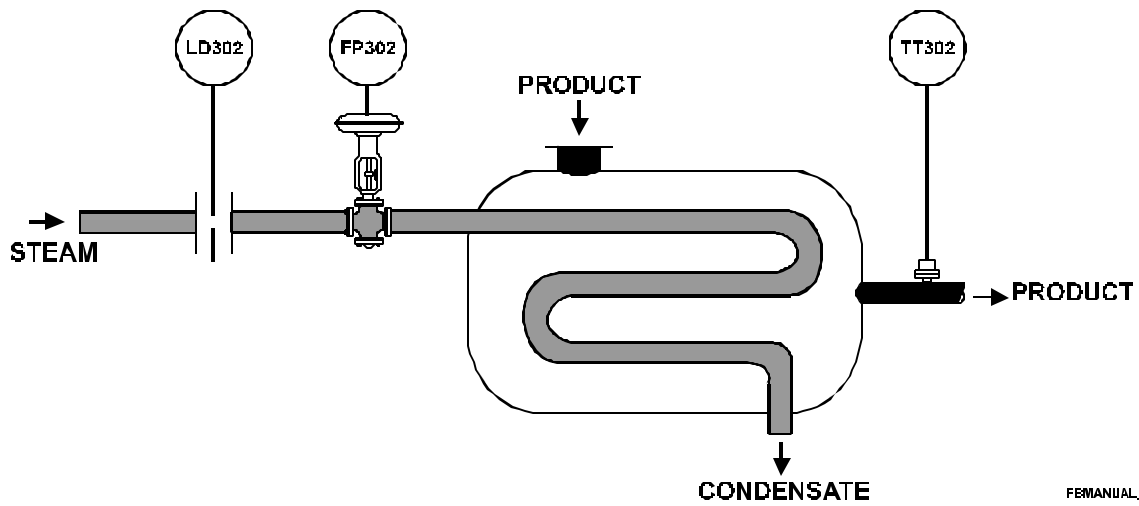
Bit	Meaning
0	Status of module in rack 14 slot 0
1	Status of module in rack 14 slot 1
2	Status of module in rack 14 slot 2
3	Status of module in rack 14 slot 3
4	
5	
6	
7	

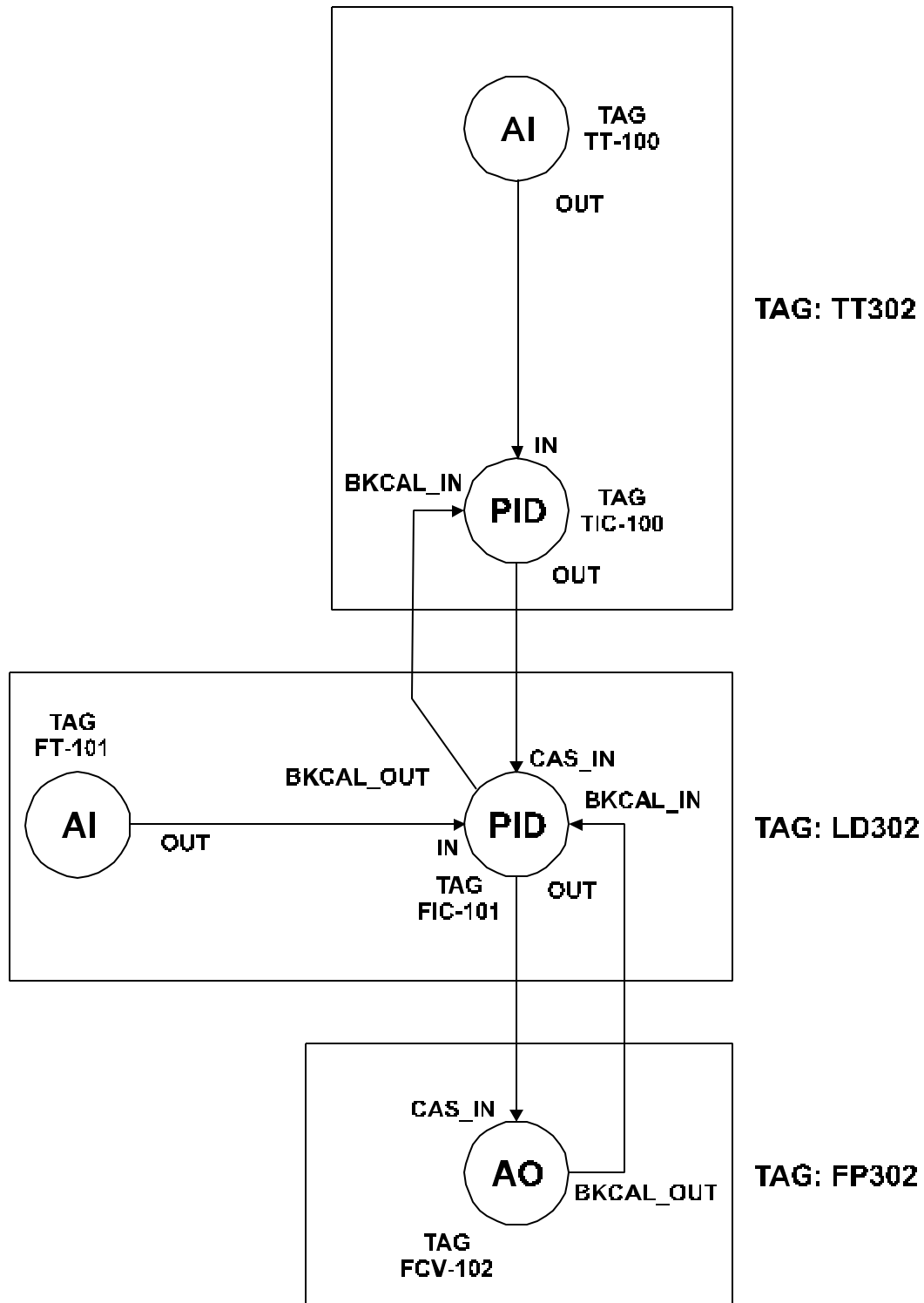
3 - EXAMPLES

SIMPLE CONTROL APPLICATION



CASCADE CONTROL



Corresponding Configuration

FBMANUAL_03

Parameterization**AI BLOCK (TT302) :**

TAG=TT-100

MODE_BLK.TARGET=AUTO

PID BLOCK (TT302) :

TAG=TIC-100

MODE_BLK.TARGET=AUTO

PV_SCALE=0 - 600 °C

OUT_SCALE=0 - 200 kg/h

AI BLOCK (LD302) :

TAG=FT-101

MODE_BLK.TARGET=AUTO

L_TYPE = Indirect, square root

XD_SCALE=0 - 200 inH2O

OUT_SCALE=0 - 200 kg/h

PID BLOCK (LD302) :

TAG=FIC-101

MODE_BLK.TARGET=CAS

PV_SCALE=0 - 200 kg/h

OUT_SCALE=0 - 100 %

AO BLOCK (FP302) :

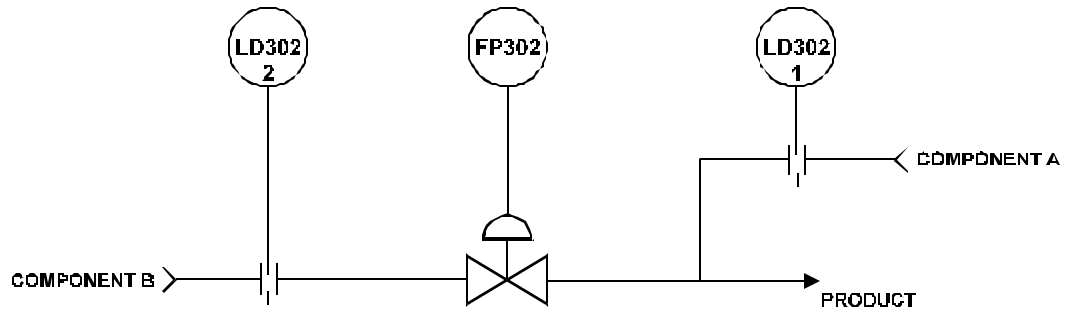
TAG=FCV-102

MODE_BLK.TARGET=CAS

PV_SCALE=0 - 100 %

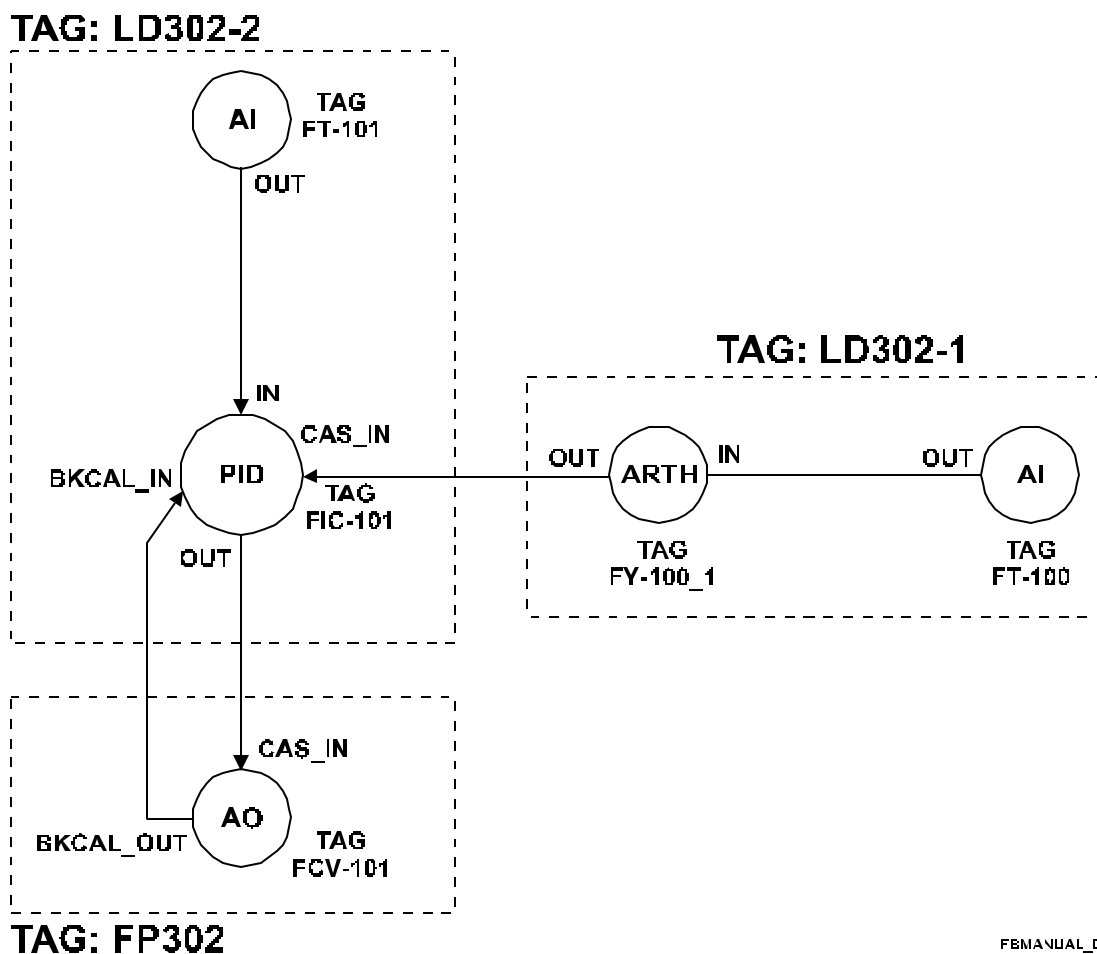
XD_SCALE=3 - 15 psi

RATIO CONTROL



FBMANUAL_04

Corresponding Configuration



Parameterization

AI BLOCK (LD302-1) :

TAG=FT-100

MODE_BLK.TARGET=AUTO

ARTH BLOCK (LD302-1) :

TAG=FY-100_1

MODE_BLK.TARGET=AUTO

ARITH_TYPE=7

GAIN = adjusted by user to desired ratio.

RANGE_LO=0

RANGE_HI=-10 (for g = 1)

AI BLOCK (LD302-2) :

TAG=FT-101

MODE_BLK.TARGET=AUTO

PID BLOCK (LD302-2) :

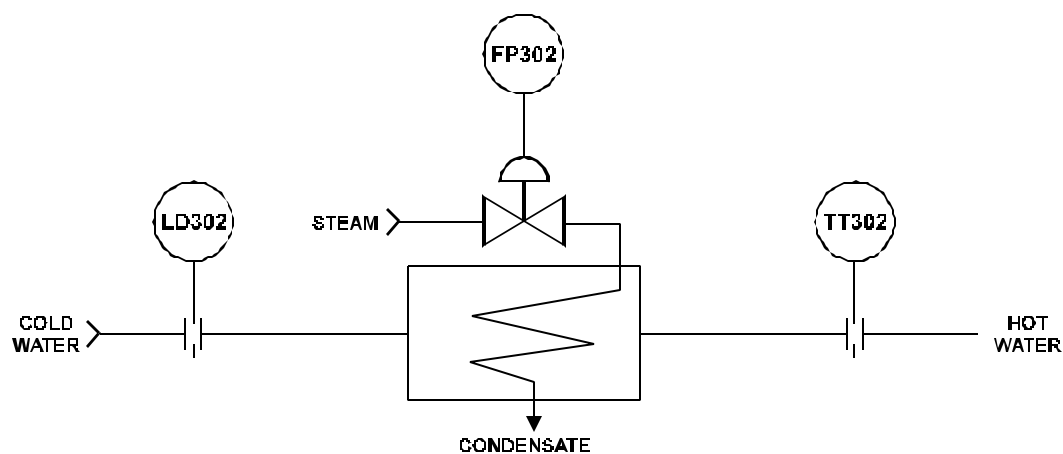
TAG=FIC-101

MODE_BLK.TARGET=CAS
PV_SCALE=0 - 200 inH2O
OUT_SCALE=0 - 100 %

AO BLOCK (FP302) :

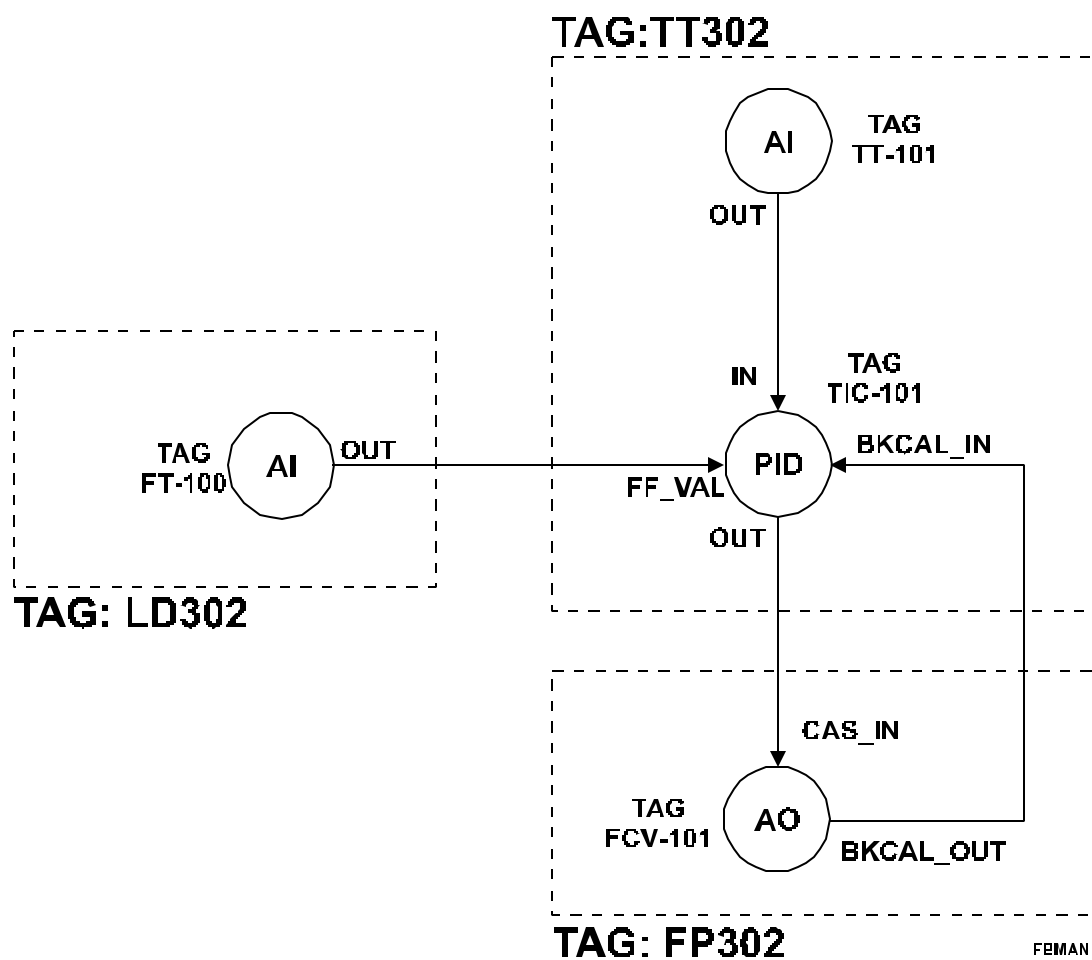
TAG=FCV-101
MODE_BLK.TARGET=CAS
PV_SCALE=0 - 100 %
XD_SCALE=3-15 psi

FEEDFORWARD CONTROL



FBMANUAL_06

Corresponding Configuration



FBMANUAL_07

Parametrization

AI BLOCK (TT302):

TAG=TT-101
MODE_BLK.TARGET=AUTO

PID BLOCK (TT302):

TAG=TIC-101
MODE_BLK.TARGET=AUTO
PV_SCALE=0 - 600 °C
OUT_SCALE=0 - 100 %
FF_SCALE=0 - 500 GAL/min
FF_GAIN=0.1

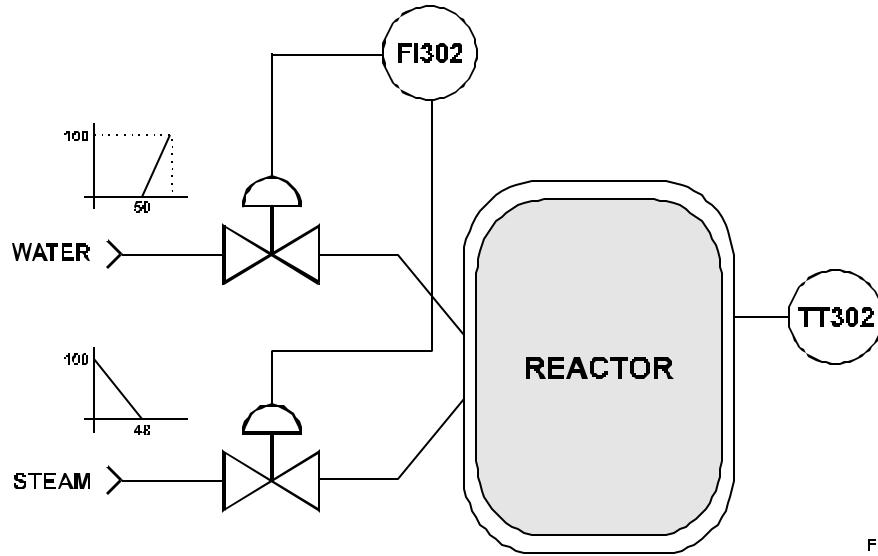
AI BLOCK (LD302):

TAG=FT-100
MODE_BLK.TARGET=AUTO
L_TYPE = Indirect, square root
XD_SCALE=0 - 125 inH2O
OUT_SCALE=0 - 500 GAL/min

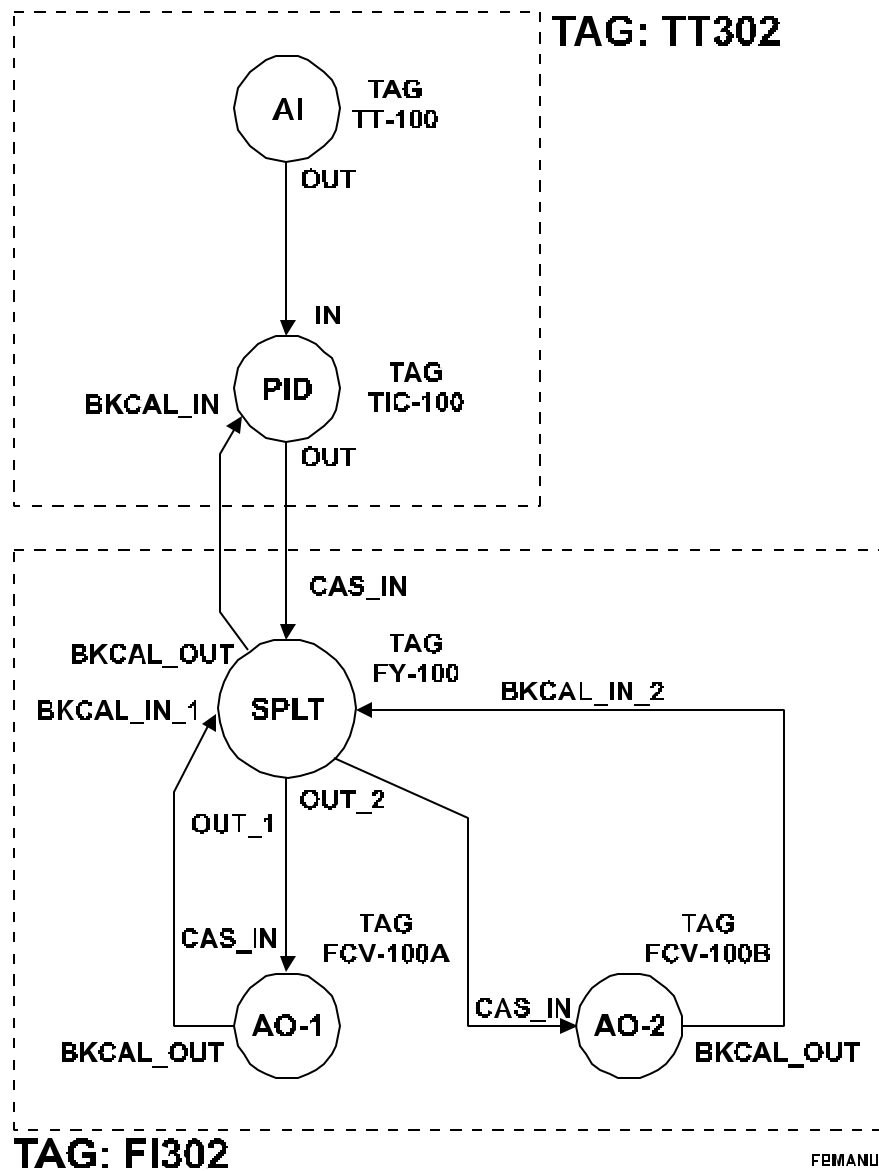
AO BLOCK (FP302):

TAG=FCV-101
MODE_BLK.TARGET=CAS
PV_SCALE=0 - 100 %
XD_SCALE=3-15psi

SPLIT RANGE CONTROL



FBMANUAL_08

Corresponding Configuration

FEMANUAL_09

Parameterization

AI BLOCK (TT302) :

TAG=TT-100

MODE_BLK.TARGET=AUTO

PID BLOCK (TT302) :

TAG=TIC-100

MODE_BLK.TARGET=AUTO

PV_SCALE=0 - 600 °C

OUT_SCALE=0 - 100 %

SPLT BLOCK (TT302) :

TAG=FY-100

MODE_BLK.TARGET=CAS

LOCK_VAL=YES

IN_ARRAY=0, 48, 50, 100

OUT_ARRAY=100, 0, 0, 100

AO 1 BLOCK (FI302) :

TAG=FCV-100A

MODE_BLK.TARGET=CAS

PV_SCALE=0 - 100 %

XD_SCALE= 4-20 mA

AO 2 BLOCK (FI302) :

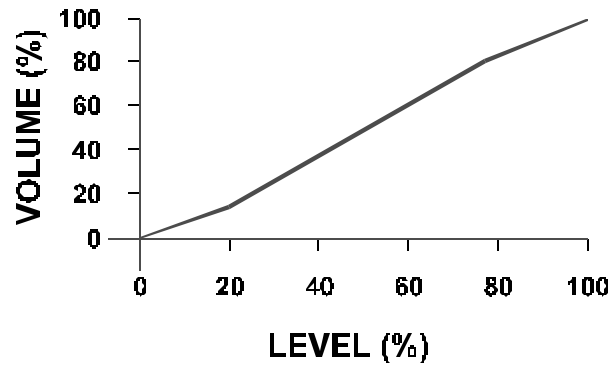
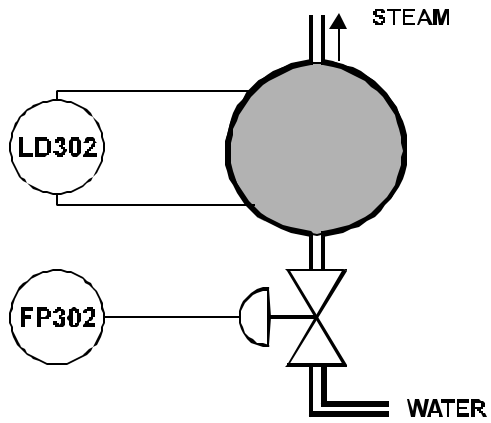
TAG=FCV-100B

MODE_BLK.TARGET=CAS

PV_SCALE=0 - 100 %

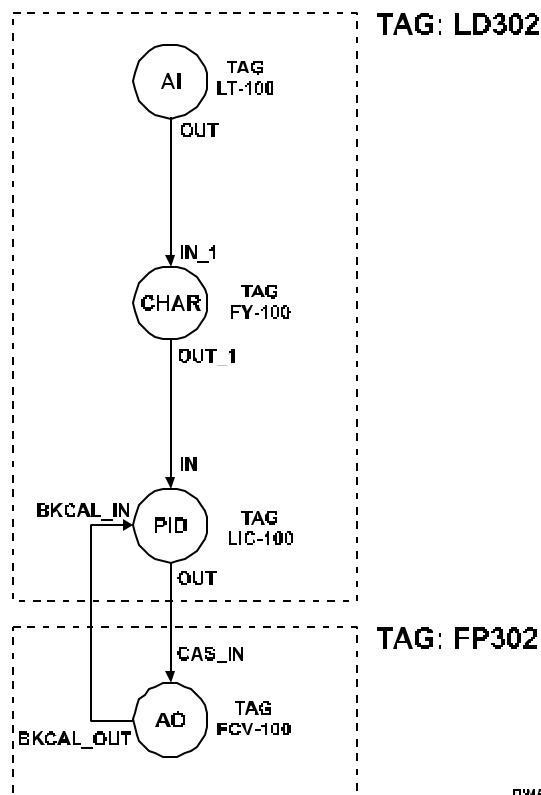
XD_SCALE= 4-20 mA

LEVEL CONTROL



FBV/ANLIA_1_1

Corresponding Configuration



FMANUAL_21

Parameterization

AI BLOCK (LD302) :

TAG=LT-100
 MODE_BLK.TARGET=AUTO

CHAR BLOCK (LD302) :

TAG=FY -100
 MODE_BLK.TARGET = AUTO
 X_UNITS= inH2O
 Y_UNITS= gal
 CURV_INPUTS=(0,40,80,100,120,160,200)
 CURV_OUTPUTS=(0, 14.23, 37.35, 50, 62.64, 85.76, 100)

PID BLOCK (LD302) :

TAG=LIC -100
 MODE_BLK.TARGET=AUTO
 PV_SCALE=0 - 100 gal
 OUT_SCALE=0 - 100%

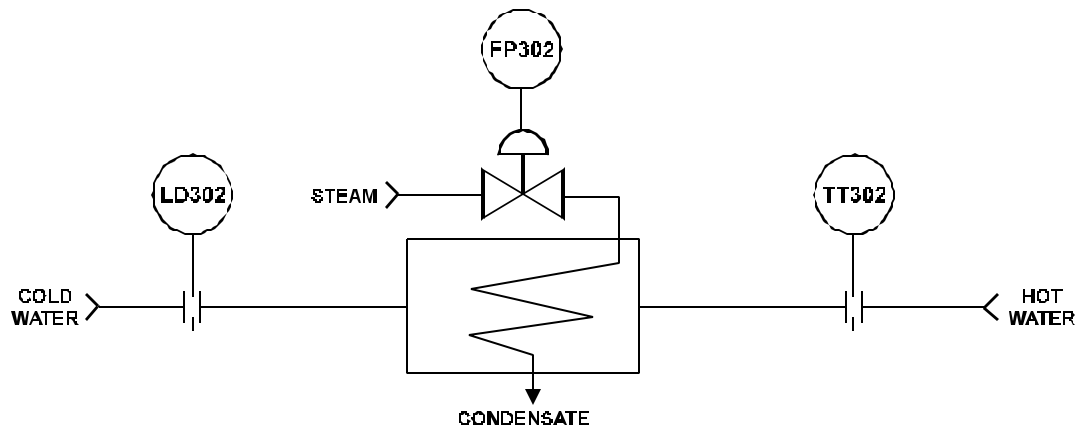
AO BLOCK (FP302) :

TAG=FCV -100

MODE_BLK.TARGET=CAS

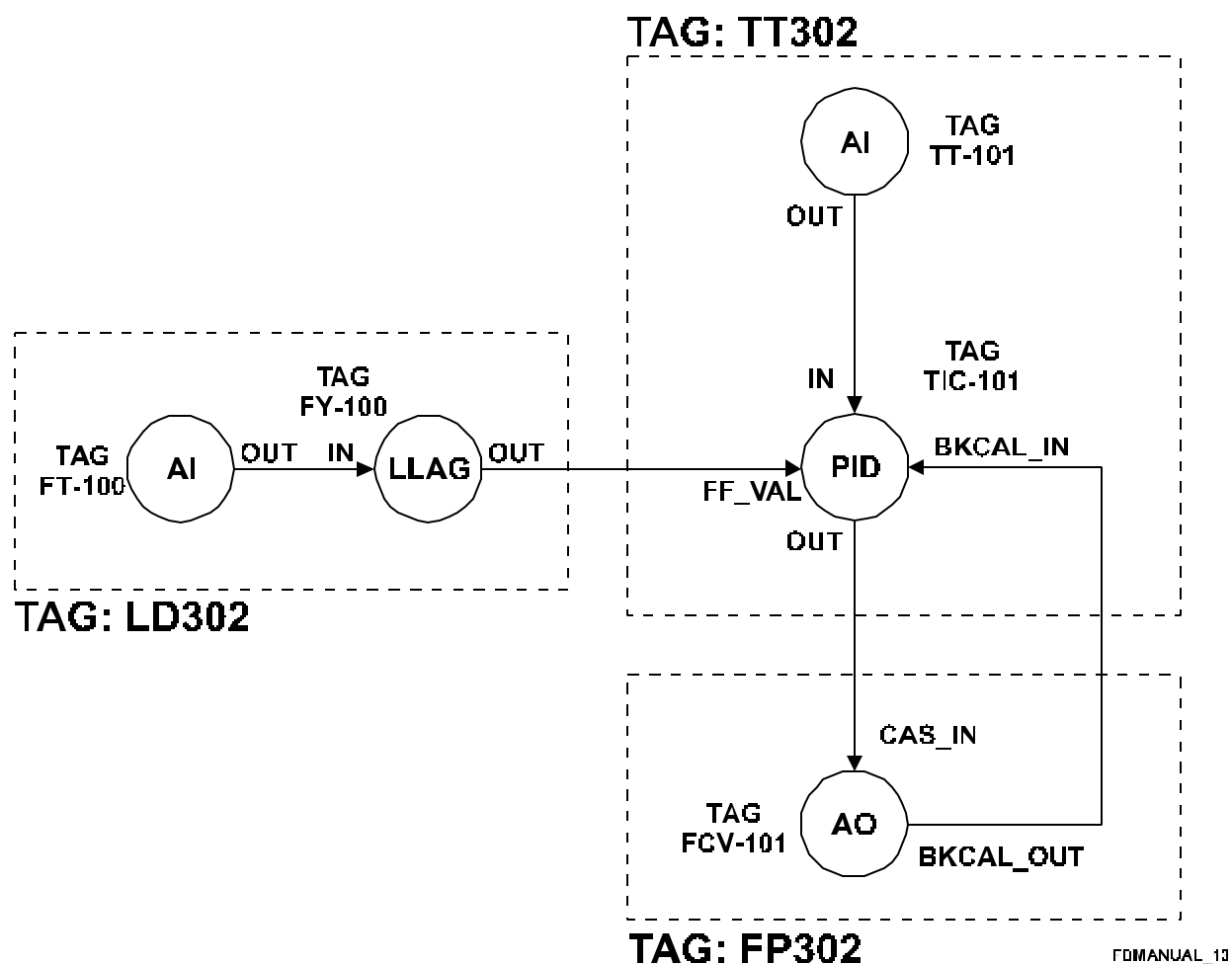
PV_SCALE=0 - 100 %

XD_SCALE= 3-15 psi

RATE CONTROL LOOP WITH LEAD-LAG

FBMANUAL_12

Corresponding Configuration



FDMANUAL_13

Parameterization

AI BLOCK (TT302) :

TAG=TT-101

MODE_BLK.TARGET=AUTO

PID BLOCK (TT302) :

TAG=TIC-101

MODE_BLK.TARGET=AUTO

PV_SCALE=0 - 600 °C

OUT_SCALE=0 - 100 %

FF_SCALE=0 - 500 GAL/min

FF_GAIN=0.1

AI BLOCK (LD302) :

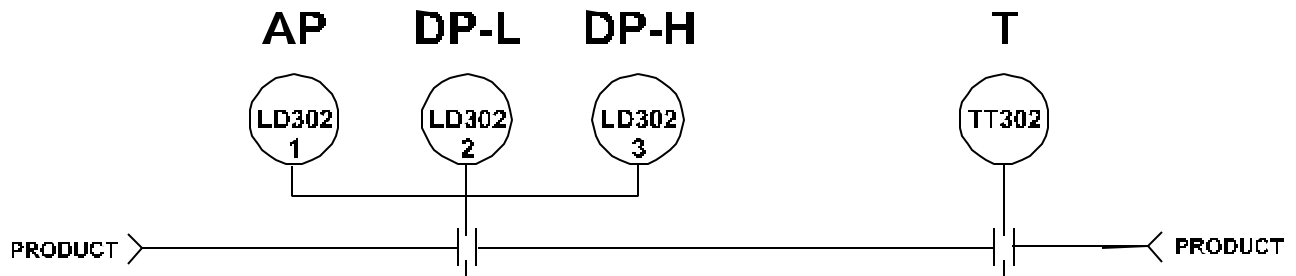
TAG=FT-100
MODE_BLK.TARGET=AUTO
XD_SCALE=0 - 125 inH2O
OUT_SCALE=0 - 500 GAL/min
L_TYPE = Indirect, square root

LLAG BLOCK (LD302):

TAG=FY-100
MODE_BLK.TARGET=AUTO
OUT_UNIT=GAL/min
LEAD_TIME=60
LAG_TIME=60

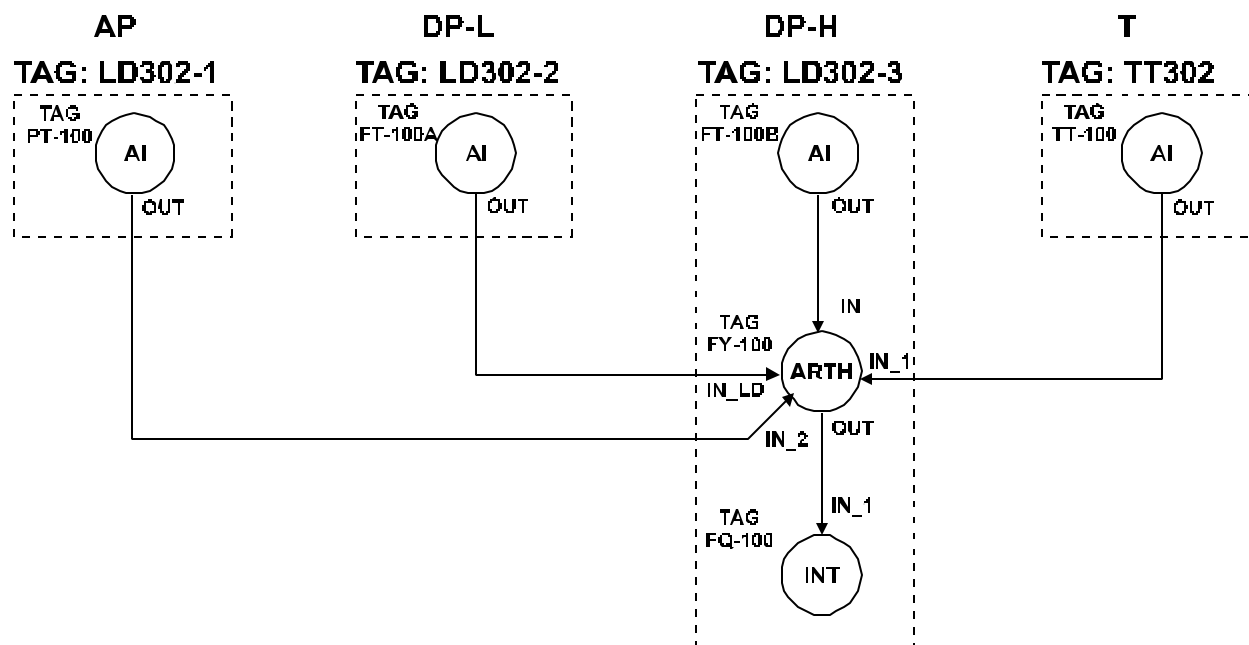
AO BLOCK (FP302) :

TAG=FCV-101
MODE_BLK.TARGET=CAS
PV_SCALE=0 - 100 %
XD_SCALE= 3-15 psi

FLOW COMPENSATION CONFIGURATION, WITH TOTALIZATION

FBMANUAL_14

Corresponding Configuration



PERMANUAL_19

Parameterization

AI BLOCK (LD302-1) :

TAG=PT-100
 MODE_BLK.TARGET=AUTO
 L_TYPE=DIRECT
 XD_SCALE.UNIT = Pa

AI BLOCK (LD302-2) :

TAG=FT-100A
 MODE_BLK.TARGET=AUTO
 XD_SCALE=0 - 20 H₂O
 OUT_SCALE=0 - 156 Cutf/min
 L_TYPE=SQR ROOT

AI BLOCK (LD302-3) :

TAG=FT-100B
 MODE_BLK.TARGET=AUTO
 XD_SCALE=0 - 200 inH₂O
 OUT_SCALE=0 - 495 Cutf/min
 L_TYPE=SQR ROOT

ARTH BLOCK (LD302-3) :

TAG=FY-100MODE_BLK.TARGET=AUTO
PV_UNIT=GAL/min
OUT_UNIT=GAL/min
ARITH_TYPE=2 (flow comp. square root – AGA3)
GAIN_IN_1 = 1
GAIN = 1
RANGE_LO=400
RANGE_HI=600
COMP_HI_LIM=+INF
COMP_LO_LIM=-INF

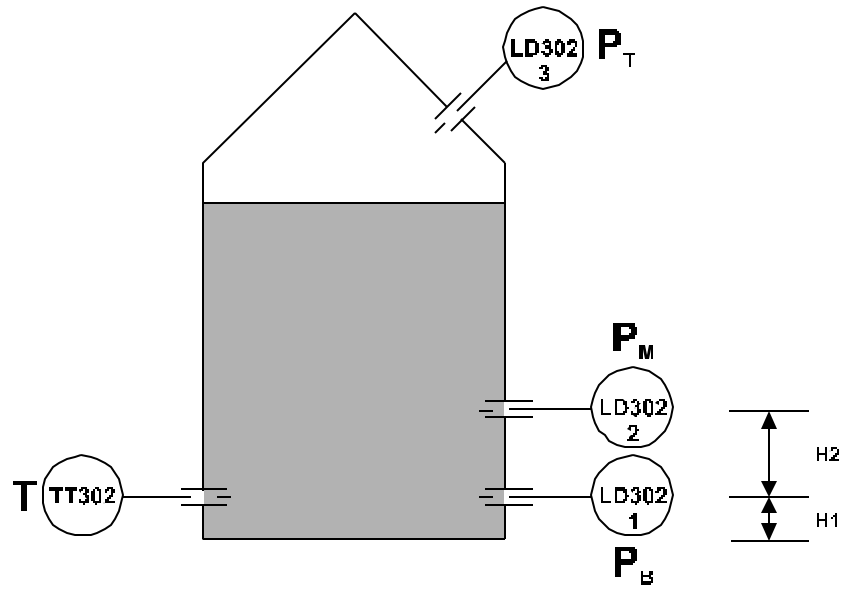
INT BLOCK (LD302-3) :

TAG=FQ-100
MODE_BLK.TARGET=AUTO
OUT_UNITS=GAL

AI BLOCK (TT302) :

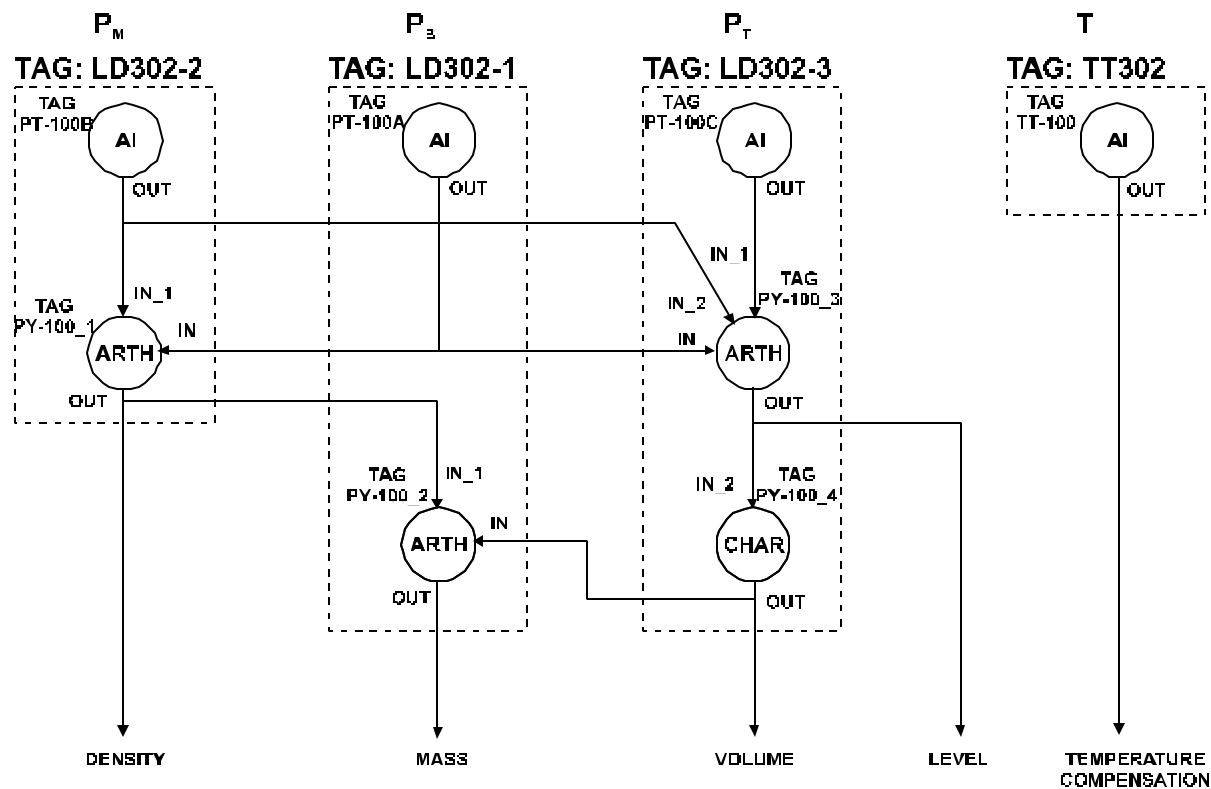
TAG=TT-100
MODE_BLK.TARGET=AUTO
XD_SCALE.UNIT=K

HYDROSTATIC TANK GAUGING



FBMANUAL_18

Corresponding Configuration



FCNANUAL_17

Parameterization

AI BLOCK (LD302-2) :

TAG=PT-100B
 MODE_BLK.TARGET=AUTO
 XD_SCALE.UNIT = Pa

ARTH BLOCK (LD302-2) :

TAG=PY-100_1
 MODE_BLK.TARGET=AUTO
 OUT_UNIT=kg/m³
 ARITH_TYPE=7 (traditional summer)
 GAIN_IN_1=1
 GAIN = 1/(H2*g)
 RANGE_LO=-20
 RANGE_HI=-10

AI BLOCK (LD302-1) :

TAG=PT-100A
 MODE_BLK.TARGET=AUTO
 XD_SCALE.UNIT = Pa

ARTH BLOCK (LD302-1) :

TAG=PY-100_2
MODE_BLK.TARGET=AUTO
PV_UNIT=m³
OUT_UNIT = Ton
ARITH_TYPE = 5 (traditional mult. div.)
GAIN_IN_2 = 1
GAIN = 1
COMP_HI_LIM=+INF
COMP_LO_LIM=-INF

AI BLOCK (LD302-3) :

TAG=PT-100C
MODE_BLK.TARGET=AUTO
XD_SCALE.UNIT = Pa

ARTH BLOCK (LD302-3) :

TAG=PY-100.3
MODE_BLK.TARGET = AUTO
PV_UNIT=mH2O
OUT_UNIT = m
ARITH_TYPE = 9 (HTG comp. level)
GAIN = H2
BIAS = H1
RANGE_LO=-20
RANGE_HI=-10

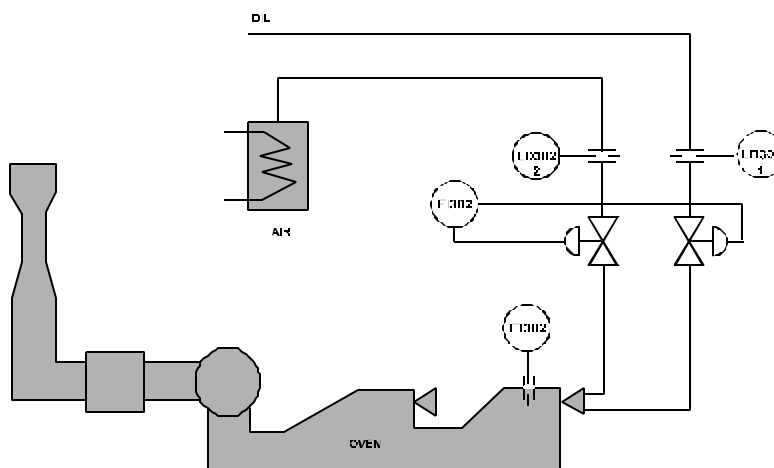
CHAR BLOCK (LD302-3) :

TAG=PY -100_4
MODE_BLK.TARGET = AUTO
X_UNITS = m
Y_UNITS = m³
CURVE_X=(0,20,40,50,70,80,100)
CURVE_Y=(0,20,40,50,70,80,100)

AI BLOCK (TT302) :

TAG=TT-100
MODE_BLK.TARGET=AUTO

COMBUSTION CONTROL WITH DOUBLE CROSS LIMITS



PERMANENT

This type of control tries to keep the air-fuel ratio strictly within the limits. A sudden change on the load would require a corresponding air and fuel variation.

The master controller supplies Setpoint values to air and fuel flow controllers while it is stabilized.

During the transitions, the air flow determines the maximum upper and lower limits that the fuel flow cannot exceed. The same occurs for the air flow, whose limits are fixed by those of the fuel flow.

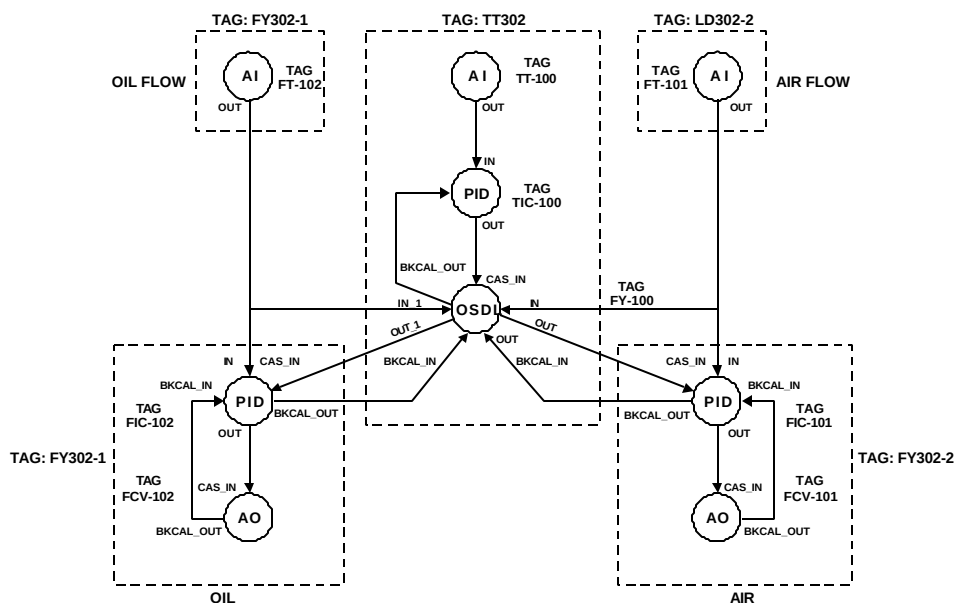
In this way, even when there is a large shift in the master signal the air/fuel ratio is maintained very close to the desired value.

The “double cross limits” prevents that the fastest variable unbalance the desired ratio. This strategy is implemented using the OSDL Block, that generates the setpoint for the air and fuel controllers based on the output of the master controller, air flow ($Q_a \rightarrow$ IN parameter) and fuel flow ($Q_c \rightarrow$ IN 1 parameter).

This configuration allows the air flow setpoint to vary just between (Qc-LO_BIAS) and (Qc+HI_BIAS) and the fuel flow setpoint to vary just between (Qa-LO_BIAS 1) and (Qa+HI_BIAS 1).

When the double crossed limit is interfered with, then an unexpected change in the consumption upsets the desired ratio and in the same way when there is a transient in the master signal of the air/fuel flow it is able to be maintained very close to the desired ratio.

Corresponding Configuration



Parameterization

AI BLOCK (LD302-1) :

TAG=FT-102
 MODE_BLK.TARGET=AUTO
 XD_SCALE=0 - 300 mm H₂O
 OUT_SCALE=0 - 100 %

AI BLOCK (LD302-2) :

TAG=FT-101
 MODE_BLK.TARGET=AUTO
 XD_SCALE=0 - 200 inH₂O
 OUT_SCALE=0 - 100 %

AI BLOCK (TT302) :

TAG=TT-100
 MODE_BLK.TARGET=AUTO
 XD_SCALE=0 - 600 °C
 L_TYPE=direct

PID BLOCK (TT302) :

TAG=TIC-100
 MODE_BLK.TARGET=AUTO
 PV_SCALE=0 - 600 °C
 OUT_SCALE=0 - 100 %

CONTROL_OPTS.Direct-acting = Reverse

OSDL BLOCK (TT302) :

TAG=FY-100
MODE_BLK.TARGET=CAS
OUT_TYPE=dynamic limiter
HI_GAIN=1
HI_BIAS=5%
LO_GAIN=1
LO_BIAS=2%
HI_GAIN_1=1
HI_BIAS_1=2%
LO_GAIN_1=1
LO_BIAS_1=5%
GAIN = 1
GAIN_1=1

PID BLOCK (FY302-1) :

TAG=FIC-102
MODE_BLK.TARGET=CAS
PV_SCALE=0 - 100 %
OUT_SCALE=0 - 100 %
CONTROL_OPTS.Direct-acting = REVERSE

AO BLOCK (FY302-1) :

TAG=FCV-102
MODE_BLK.TARGET=CAS
PV_SCALE=0 - 100 %
XD_SCALE= 0 – 100%

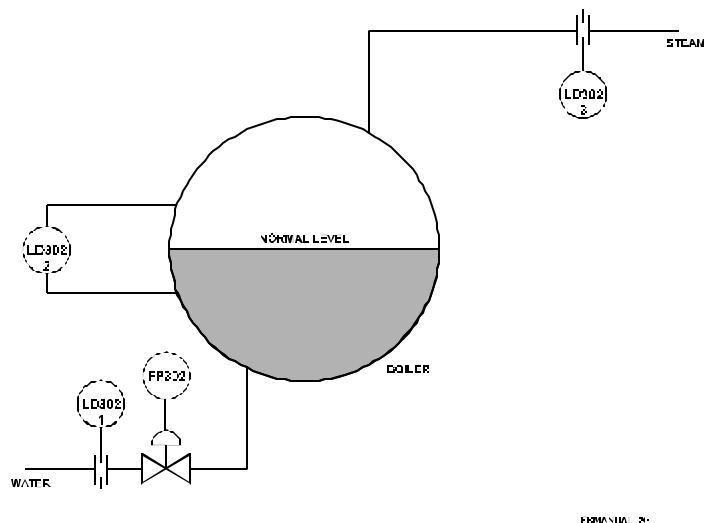
PID BLOCK (FY302-2) :

TAG=FIC-101
MODE_BLK.TARGET=CAS
PV_SCALE=0 - 100 %
OUT_SCALE=0 - 100 %
CONTROL_OPTS.Direct Acting=REVERSE

AO BLOCK (FY302-2) :

TAG=FCV-101
MODE_BLK.TARGET=CAS
PV_SCALE=0 - 100 %
XD_SCALE= 0 – 100%

3 ELEMENT BOILER LEVEL / FEEDWATER CONTROL



This control loop uses feed forward control combined with cascade control. In this case the feed forward steam flow correction is done for steam flow and the feedback is done through the transmitter and the level controller LIC-100, whilst the feed water flow is maintained by the secondary cascade control loop of water.

In this loop the drum level controller LIC-100 provides the setpoint for the feedwater controller FIC-100 in cascade. Any disturbance in feed water flow is corrected by a feed forward arrangement in FIC-100. By connecting FT-101 to the FF_VAL input of FIC-100, any change in feed water flow adjusts the FIC-100 setpoint directly. The FF_SCALE is set -100 to +100 % to provide a fixed 50 % bias, giving a 50 % setpoint when load and manipulated flow are perfectly matched.

AI BLOCK (LD302-3):

TAG=FT-101
MODE_BLK.TARGET=AUTO
XD_SCALE=0 - 9500 mmH2O
OUT_SCALE=0 - 150 Ton/hr
L_TYPE = Indirect, Square Root

INT BLOCK (LD302-3):

TAG=FQ-101
MODE_BLK.TARGET= AUTO
TIME_UNIT1=HourS
OUT_UNITS = Ton

AI BLOCK (LD302-1):

TAG=FT-100
MODE_BLK.TARGET=AUTO
XD_SCALE=0 - 3500 mmH2O
OUT_SCALE=0 - 150 m³/hr
L_TYPE = Indirect, Square Root

INT BLOCK (LD302-1):

TAG=FQ-100
MODE_BLK.TARGET = Auto
TIME_UNIT1=Hours
OUT_UNITS=m³

PID BLOCK (FP302):

TAG=FIC-100
MODE_BLK.TARGET=CAS
PV_SCALE=0 - 150 m³/hr
OUT_SCALE=0 - 100 %
CONTROL_OPTS.Direct Acting = Reverse
FF_SCALE = -100 to +100 %
FF_GAIN = 1

AO BLOCK (FP302):

TAG=FCV-100
MODE_BLK.TARGET=CAS
PV_SCALE=0 - 100 %
XD_SCALE = 3-15 psi